

# **TDK-dolgozat**

Dénes Botond

# Vízközeli hulladéklerakók megbízható detektálása multispektrális műholdfelvételek segítségével

EÖTVÖS LORÁND TUDOMÁNYEGYETEM

INFORMATIKAI KAR

PROGRAMOZÁSELMÉLET ÉS SZOFTVERTECHNOLÓGIAI TANSZÉK



*Szerző:*

Dénes Botond  
programtervező informatikus MSc  
2. évfolyam

*Témavezető:*

Cserép Máté  
egyetemi tanársegéd

Budapest, 2024

# Tartalomjegyzék

<b>1. Bevezetés</b>	<b>3</b>
1.1. A kutatólabor eddigi eredményei . . . . .	4
1.2. Kutatási cél . . . . .	4
1.3. A dolgozat felépítése . . . . .	5
<b>2. Kapcsolódó kutatások</b>	<b>6</b>
2.1. Hulladékdetektálási módszerek . . . . .	6
2.2. Döntési fa . . . . .	12
2.3. Random Forest . . . . .	13
2.4. PlanetScope . . . . .	14
<b>3. Betanítás</b>	<b>16</b>
3.1. Tanító adatok . . . . .	16
3.2. Használt sávok és indexek . . . . .	18
3.3. Tanítási paraméterek . . . . .	20
3.4. Felvételek normalizálása . . . . .	21
3.5. Főkomponens analízis (PCA) . . . . .	22
3.6. Nyári és téli adatokra való lebontás . . . . .	25
<b>4. Verifikáció</b>	<b>26</b>
4.1. A teszthalmaz . . . . .	26
4.2. Teljesítmény mérése . . . . .	28
4.3. Főkomponens analízis teljesítménye . . . . .	29
4.4. Vízmaszkolás . . . . .	32
4.5. Nyári adatokon betanított modell teljesítménye . . . . .	33
4.6. Normalizálás tesztelése . . . . .	34
4.7. Empirikus validáció . . . . .	35
<b>5. Megvalósítás és alkalmazás</b>	<b>39</b>

5.1.	A meglevő alkalmazások bővítése . . . . .	39
5.1.1.	Az asztali alkalmazás bővítése . . . . .	39
5.1.2.	A szerveralkalmazás bővítése . . . . .	39
5.2.	A Tiszta-Tisza alkalmazás . . . . .	40
5.2.1.	A képfeldolgozás gyorsítása . . . . .	41
5.3.	Közös metszet . . . . .	45
<b>6.</b>	<b>Összefoglalás és eredmények</b>	<b>47</b>
6.1.	A kutatás során elért eredmények . . . . .	47
6.2.	A kutatás kihívásai . . . . .	48
6.3.	További lépések . . . . .	48
	<b>Köszönetnyilvánítás</b>	<b>49</b>
	<b>Irodalomjegyzék</b>	<b>49</b>
	<b>Ábrajegyzék</b>	<b>55</b>
	<b>Táblázatjegyzék</b>	<b>57</b>

# 1. fejezet

## Bevezetés

Napjainkban nagy kihívást jelentenek a hulladékkal szennyezett területek: ezek kárt okoznak a környező élővilágban, illetve emberek számára is egészségügyi kockázatot jelentenek. [1]. Ezért kialakultak olyan önkéntes szervezetek, mint a PET Kupa, akik hulladékgyűjtéssel foglalkoznak elsősorban a magyarországi folyók mentén, de szomszédos országokra is kiterjed a figyelmiük. Az egyik nagy kihívás a szemétgyűjtésben a hulladékkal szennyezett területeknek a hatékony megtalálása. Sok erőforrást igényel a szemetlerakók megtalálása a folyók mentén, hiszen sokszor járművel kell valakinek végig haladnia egy hosszabb területen, azért, hogy felmérje, hogy hol van hulladék. A folyók árterén elhelyezett hulladék még nagyobb problémát jelent, hiszen áradás idejében a szemetet elmosa a víz és ez a folyó további szakaszaira lesz szétszórva miközben nagy károkat okoz a folyó élővilágának, illetve szennyezi a folyó vizet [2, 3]. Emiatt szükségünk van olyan eszközökre, melyekkel hamar lehet detektálni a szennyezett területeket, hogy ezeket minél hatékonyabban meglehessen tisztítani. Az ELTE IK Térinformatikai Labor és a PET Kupa együttműködésében olyan eszközöket fejlesztünk, melyek automatikusan képesek lesznek hulladékot detektálni a folyók mentén.

A dolgozatomban bemutatok egy Random Forest modellt [4], mely a kutatólaborban már korábban kifejlesztett módszerekre épül [5]. A bemutatott modell javít a korábbi megoldás *false positive* arányain, miközben tovább is képes hulladékot detektálni. A modell eredményei integrálásra kerülnek a Tiszta Tisza webalkalmazásba [6]<sup>1</sup>, ahol több napon keresztül történő detektálás eredménye lesz összesítve és megjelenítve a felhasználók számára.

---

<sup>1</sup>A Tiszta-Tisza webalkalmazás hivatalos weboldala a hivatkozott címen található, de a dolgozatomban említett fejlesztések jelenleg a <https://gis.inf.elte.hu/tiszta-tisza/> oldalon érhetők el.

A kutatás hozzáadott terméke egy olyan adathalmaz, mely alkalmas más hulladékdetektálási modellek betanítására is. Az adathalmaz elsősorban szárazföldi romániai hulladéklerakókról készített PlanetScope műholdfelvételeket tartalmaz, melyek kézzel voltak annotálva. Az adatok georeferálva vannak, így ezeket könnyen meg lehet vizsgálni, illetve ki lehet egészíteni. A tanítóadatok hatékony feldolgozása érdekében bemutatok egy módosítást a meglevő index-számoló algoritmusban, mellyel lényegesen csökkentem a felvételek feldolgozási idejét.

Kutatásomban a modell teljesítményének javítását további módszerekkel, így például főkomponens analízissel, képnormalizálással, vízmaszkolással, illetve az annotált felvételek évszakokra bontásával specifikus modellek készítésével is vizsgálom.

## 1.1. A kutatólabor eddigi eredményei

A ELTE IK Térinformatikai Kutatólaborában már betanításra került egy Random Forest modell, mely folyómenti hulladék detektálására alkalmas. Az én dolgozatomban, a kutatólabor meglevő tudására építve, továbbfejlesztem ezt a modellt, hogy kevesebb *false positive*-al találja meg a szeméttel szennyezett területeket. Továbbra implementálásra került egy szerveralkalmazás, mely minden nap a Planet szervereiről letölti a legfrissebb felvételeket a vizsgált területekről, és lefuttatja ezeken a képeken az akkori modellt. Ezen felül készült egy webalkalmazás is, ami erről a szerverről letölti az eredményeket, és megjeleníti ezeket, összehasonlításra. A kutatólabor rendelkezik egy asztali alkalmazással is, mellyel hatékonyan elő lehet állítani tanító adatokat. A kutatásom elősegítéséhez ezeket az alkalmazásokat használtam, illetve bővítettem az 5.1 fejezetben leírtak szerint.

## 1.2. Kutatási cél

A cél az, hogy a kutatás során szerzett modell megbízhatóan detektáljon hulladéklerakókat általánosan folyók mentén. Ehhez a *false positive* arányok minél kisebbek kell legyenek, míg a *true positive* arányok minél nagyobbak. Ugyanakkor nem jelent ugyanakkora problémát egy *false negative*, mint egy *false positive*, mivel a *false positive* eredmények fölöslegesen rossz irányba küldhetik a folyómentő csapatot. A kutatólabor 2023-as cikkében bemutatott modell (továbbiakban meglevő modell vagy régi modell) [5] egyik problémája a nagy *false positive* arányok voltak. Általában a modellnek leginkább az utak, épületek okoznak problémát. Ez annak köszönhető, hogy a hulladék, a törmelék, az épü-

letek, és a föld nagyon hasonló spektrális értékekkel rendelkeznek a használt sávokon. A modell a pusztazámori hulladéklerakóról, illetve a kiskörei víztárolóról szerzett adatokkal volt betanítva. Ezért a következő lépés több adaton betanítani a modellt, nagy figyelmet fektetve arra, hogy az adathalmaz tartalmazzon bőven utakat, épületeket, és más adatokat, amik hasonlítanak a hulladékra.

### **1.3. A dolgozat felépítése**

A 2. fejezetben bemutatásra kerülnek a hulladékdetektálás témaját feldolgozó kutatások, illetve bemutatom azokat a technikai és számolási eszközöket, melyek lényeges szerepet vállalnak a kutatásomban. A 3. fejezetben részletezem a Random Forest modell betanításához előállított adatokat, a modell betanítási paramétereit, illetve megvizsgálok különböző adatfeldolgozási módszereket, ilyen például a főkomponens analízis, képnormálizálás, vízmaszkolás, azzal a céllal, hogy tovább javítsak a modell teljesítményén. A 4. fejezetben tárgyalom a modell tesztelésének és validálásának módját, illetve a tesztadatok megválasztásának módját, motivációját. Az 5. fejezetben bemutatom a kutatást lényegesen előresegítő szoftveres fejlesztéseket, illetve azt, hogy a kutatás eddigi eredményei miként vannak integrálva a Tiszta-Tisza alkalmazásba. A 6. fejezetben összefoglalom a kutatás eredményeit, és ezek alapján tárgyalom a kutatás további lehetséges haladásait.

## **2. fejezet**

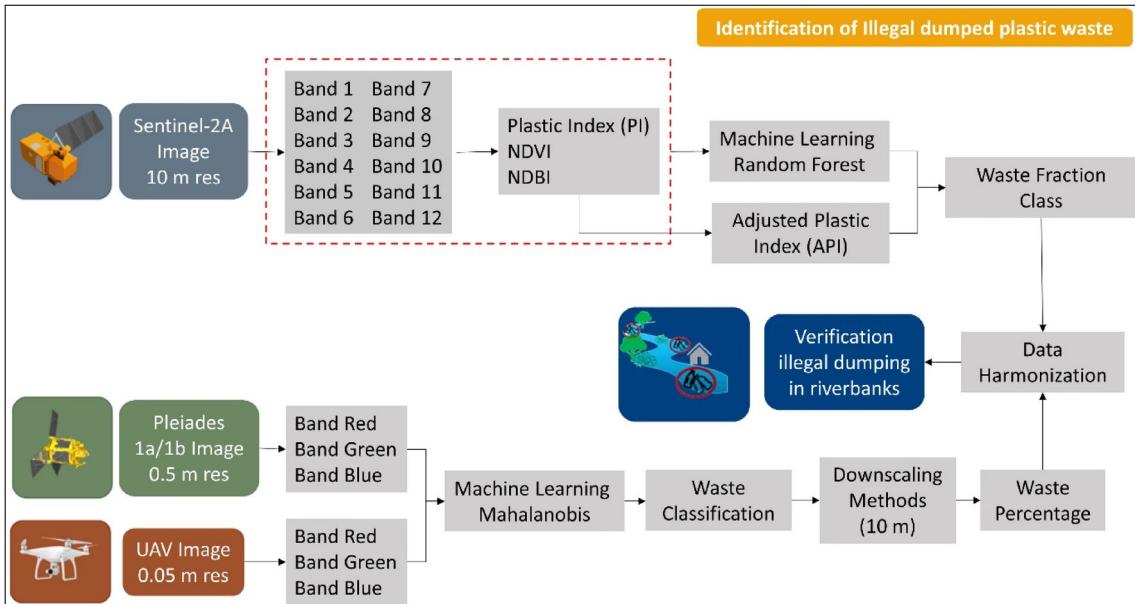
# **Kapcsolódó kutatások**

### **2.1. Hulladékdetektálási módszerek**

A hulladékdetektálás téma viszonylag új, és számos fejlemény történt az évek során, tekintve, hogy a légi és műholdfelvételek minősége is nőtt. Emiatt kevés olyan kutatás létezik, mely pontosan azzal a problémával foglalkozik, amivel ebben a dolgozatban foglalkoztam, de az érintőleges kutatások is olyan módszereket, ötleteket mutatnak be, melyeket érdemes megfontolni ebben a kutatásban is. Az én kutatásom az ELTE IK Térinformatikai kutatólabor munkájára épül [5], ahol egy Random Forest modell került betanításra, hulladékdetektálás céljából. A kutatásban PlanetScope és Sentinel-2 műholdfelvételeket használtak. Ez a cikk rakta le az alapjait az én kutatásomnak is, melyben ezeken az eredményeken javítok. A cikkben további lehetséges munkaként említésre kerül a modell több adattal való tanítása, illetve a képfeldolgozás gyorsítása. A kutatásom minden feladattal foglalkozik. A dolgozatomban csak a PlanetScope felvételek kerülnek felhasználásra, mivel a magasabb felbontású felvételek könnyebben lehetővé teszik a tanító adatok előállítását, hiszen jobban lehet látni a hulladéklerakókat rajtuk.

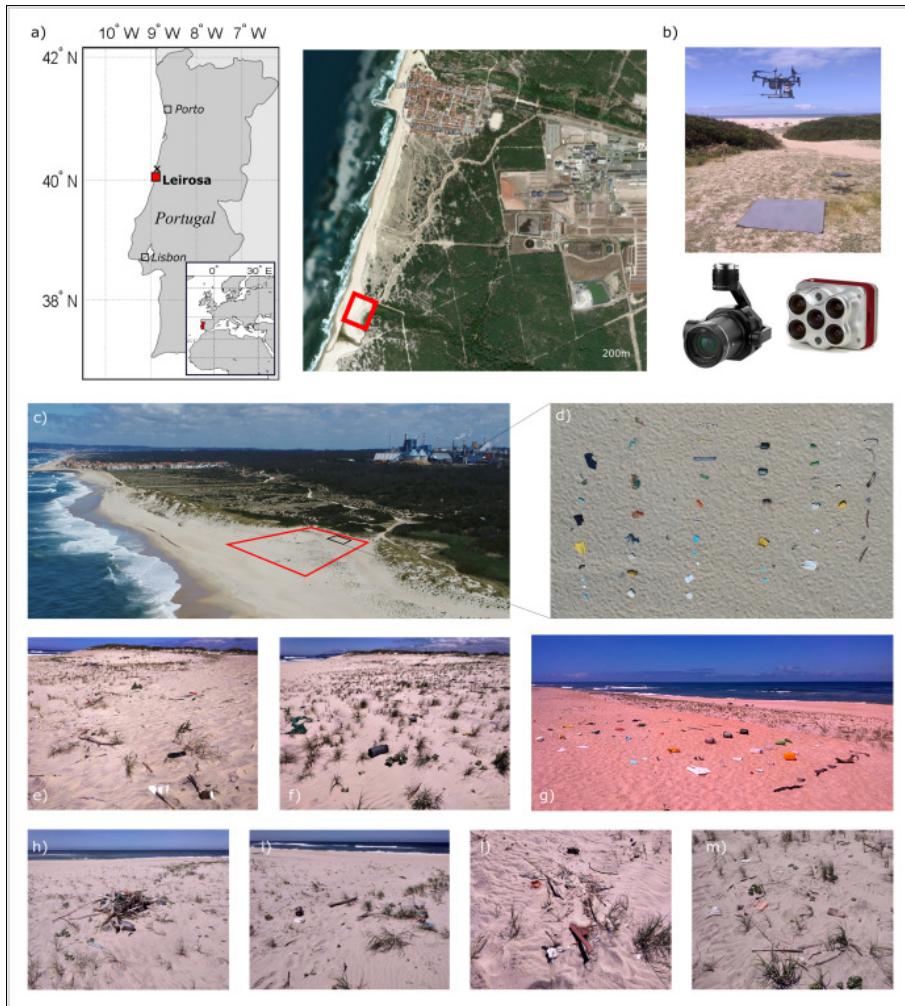
Sakti és tsai. Sentinel-2 műholdfelvételeken tanítottak be egy Random Forest modellt azzal a céllal, hogy egy indonéziai folyóban detektáljanak hulladékot [7]. A cikkben bevezetik az "Adjusted Plastic Index"-et, mellyel a vegetáció, föld és épületek közötti zajt csökkentik. Ennek az indexnek a kiszámításához a Sentinel-2 műhold piros, közeli infravörös (NIR), illetve rövid hullámhosszú közeli infravörös (SWIR) sávokat használták fel. Validációjának Pleiades műholdképeket és drónfelvételeket klasszifikáltak Mahalanobis távolság gépi tanulási módszerrel (2.1 ábra). A módszer növényzeten és vízen rendre 88%, illetve 85% pontosságot ért el és épületeken, törmeléken és földön rendre 62%, 53%,

illetve 21% pontossággal tudta a hulladékot detektálni. A cikk szerint az utóbbi három adattípuson azért visszafogottabbak az arányok, mert a spektrális értékei az épületeknek, a földnek és a törmeléknek nagyon hasonlítanak.



2.1. ábra. Hulladékdetektálás Adjusted Plastic Index, Random Forest és Mahalanobis távolság segítségével [7].

Gonçalves és Andriolo Spectral Angle Mapping módszert alkalmaztak multispektrális drónfelvételeken, egy Portugál tengerparton [8]. A célja a kutatásnak az volt, hogy a tengerpartra kimosott hulladékot detektálják és klasszifikálják. A módszer alkalmazásához referencia értékeket állítottak elő úgy, hogy elhelyeztek különböző anyagokból álló hulladékot a homokba, és ezekről drónfelvételt készítettek (2.2 ábra). Ezzel a módszerrel képesek voltak detektálni és klasszifikálni nem csak homok fölött található hulladékot, hanem a homokban félig elásott hulladékot is. A 472 kézzel előállított tesztadatból volt a 268 *true positive* (57%-a az összes adatnak), 96 volt a *false positive* és 204 volt a *false negative*.

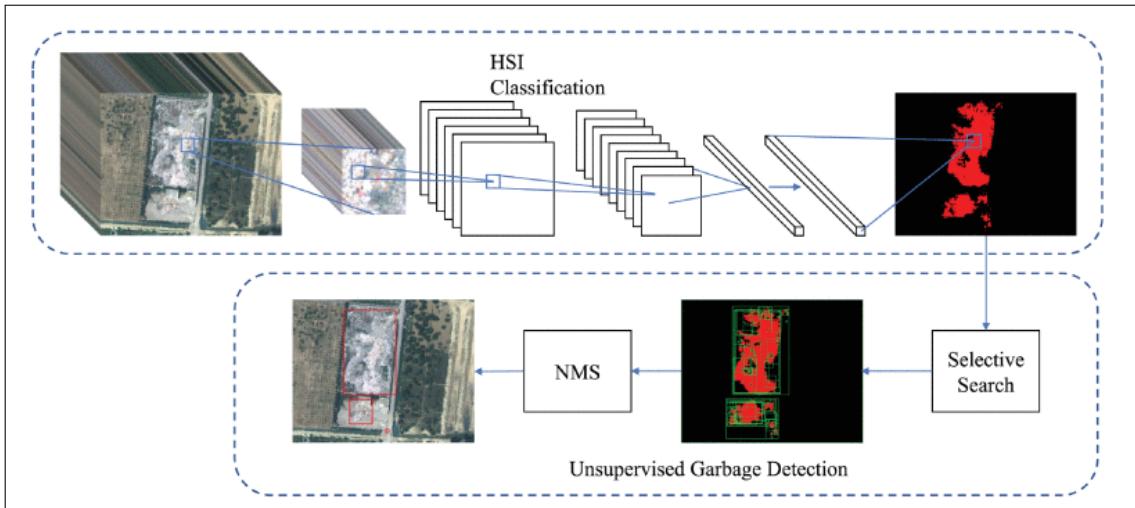


2.2. ábra. Spectral angle mapping referencia adatainak előállítása [8].

Lanorte és tsai. mezőgazdasági hulladékdetektálásra használtak egy Support Vector Machine modellt, Landsat 8 műholdfelvételeken [9]. A szenzor Kék, Zöld, Piros, NIR, SWIR 1, SWIR 2 és CIRRUS sávját használták a tanító adatok és tesztadatok előállítására. Ezután véletlenszerűen szétválasztották az adatokat tanító adatokra és tesztadatokra. A következő osztályokra bontották az adatokat: hálók, műanyag takarók, föld, növényzet, gyümölcsöskert, olajfás kert, város, fa, fás föld. A modell a tesztadatokat összességében 94%-os pontossággal tudta klasszifikálni, ahol a legrosszabb arányokat az olajfás kert érte el 77.78%-os pontossággal.

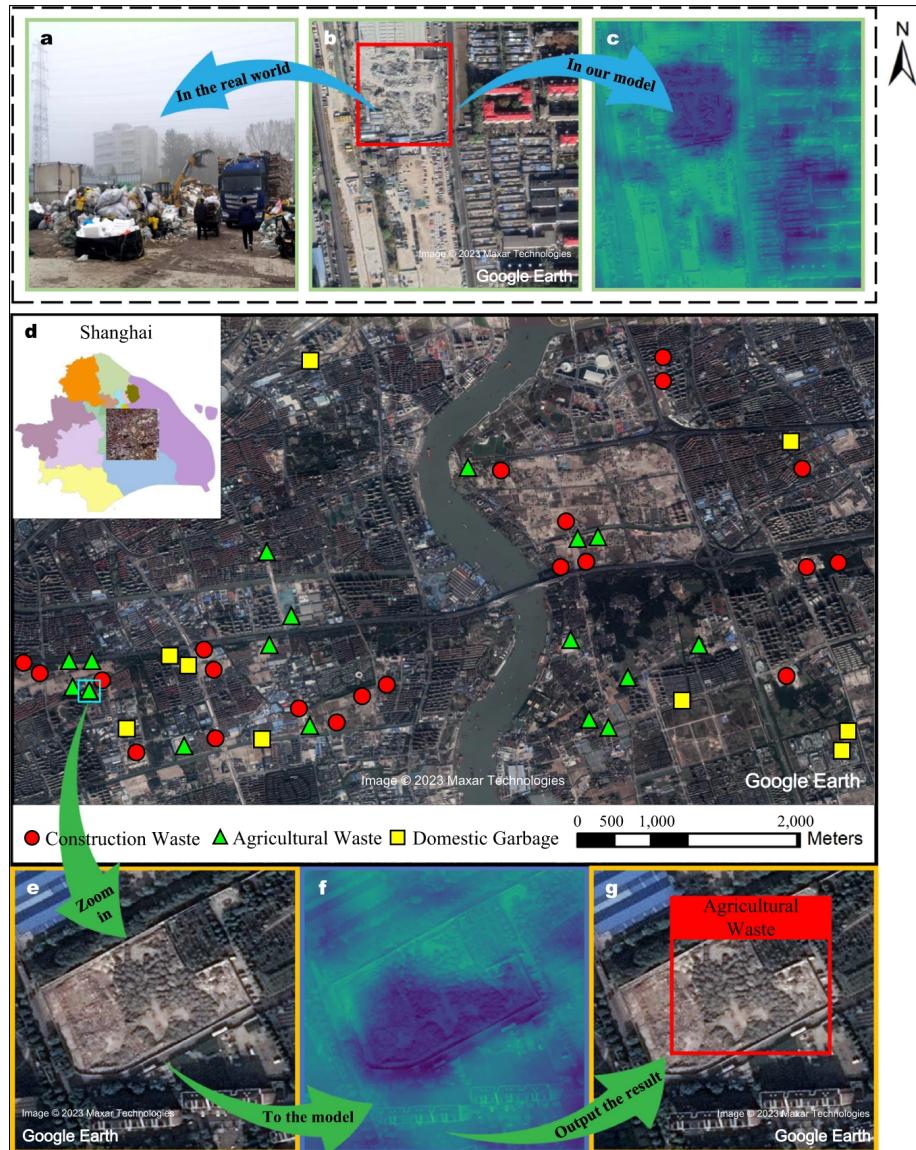
Zeng és tsai. a hiperspektrális adatokon tanítottak be egy felügyelt (Multi-Scale Convolutional Neural Network vagy MSCNN) és egy felügyeletlen gépi tanulási módszert (None Maximum Suppression vagy NMS) [10]. A működési elv az, hogy a felügyelt módszerrel klasszifikálják a hulladékkal szennyezett területeket, míg a felügyeletlen módszerrel megbecsülik a hulladékkal szennyezett terület helyét és terjedelmét (2.3 ábra). Az

MSCNN 99.96% átlagos pontossággal tudta detektálni a szemetet.



2.3. ábra. Az MSCNN és NMS működési elve [10].

Sun és tsai. megvizsgálják azt, hogy mennyivel lesz hatékonyabb a hulladékkezelés a manuális vizsgálat helyett, ha egy mély tanulás alapú hulladékdetectálási módszert használnak magas (0.3m-1m per pixel) felbontású műholdfelvételeken [11]. Négy osztályba bontják a hulladékot: háztartási hulladék, mezőgazdasági hulladék, építkezési hulladék, lefedett hulladék. A modell képes 98%-át detektálni a hulladéklerakóknak a teszthalmazban, illetve az automata detektálás segítségével több, mint 96.8%-al csökkentették azt az időt, amit a hulladéklerakók detektálására töltöttek a szakemberek. Ráadásul a modell könnyen telepíthető egy laptopra, és 30 másodperc alatt le tudja futtatni a modellt a 162 négyzetkilométeres tesztadathalmazon. A betanításhoz minden össze 2500 hulladéklerakót annotáltak világszerte, kézzel, 4800 négyzetkilométeren keresztül. A 2.4 ábrából látható a módszer működési elve: a modell a spektrális értékek mellett a hulladéklerakó alakját is figyelembe veszi. A megfelelő magabiztoságú területek hulladéklerakóként lesznek címkezve. A modellt betanító és validáló kód, illetve az adathalmaz, mellyel az eredmények reprodukálhatóak, elérhetők a cikkből.



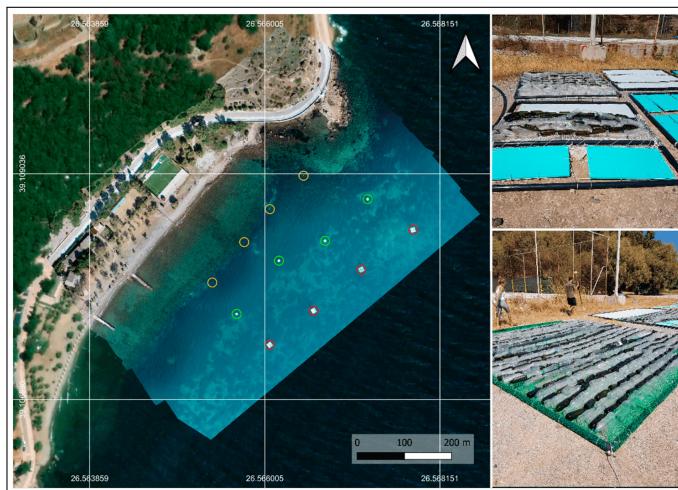
2.4. ábra. A BCA-Net működési elve [11].

Torres és Fraternali egy konvolúciós neuronhálót használnak, mely 94.5%-os átlagos pontossággal és 88.2%-os F-Score-al rendelkezik [12]. A tanító adathalmaz 3000 ortofotóból áll, melyek 20 cm per pixel felbontással rendelkeznek és a vörös, kék és zöld tartományokat tartalmazzák. A felvételeket szakértők annotálták, kézzel. A modell a ResNet50 [13] hálóra épül és a Feature Pyramid Network [14] architektúrával van kiegészítve. A modell figyelembe veszi a hulladéklerakó alakját és kontextusát, mely segít a hulladékdetektálás pontosságának a növelésében.

Page és tsai. két Random Forest algoritmust tanítottak be kerekek és műanyagok detektálására, Skóciában [15]. A kutatásukban a Copernicus Sentinel-1 és Sentinel-2 multispektrális műholdfelvételeit használták. A tesztelés során a modellek 211 kerék és műanyag alapú hulladéklerakót találtak, rendre 87.5% és 84% pontossággal. A kutatásukban

az NDVI, SAVI, illetve NDWI2 indexek kerülnek tárgyalásra és kiszámításra.

Taggio és tsai. tengeri hulladékdetectálással foglalkoztak Görögország területén levő tengerparton: egy felügyelt (Light Gradient Boosting Model) és egy felügyeletlen (K-Means) gépi tanulási algoritmust alkalmaztak PRISMA hiperspektrális műholdfelvételen, melyel átlagosan 96%-os pontosságot értek el [16]. A tanító adatokat kontrollált környezetben állították elő (2.5 ábra): különböző környezeteket szimulálva vízre helyeztek különböző anyagokból álló hulladékot, melyekről műholdfelvételek készültek.

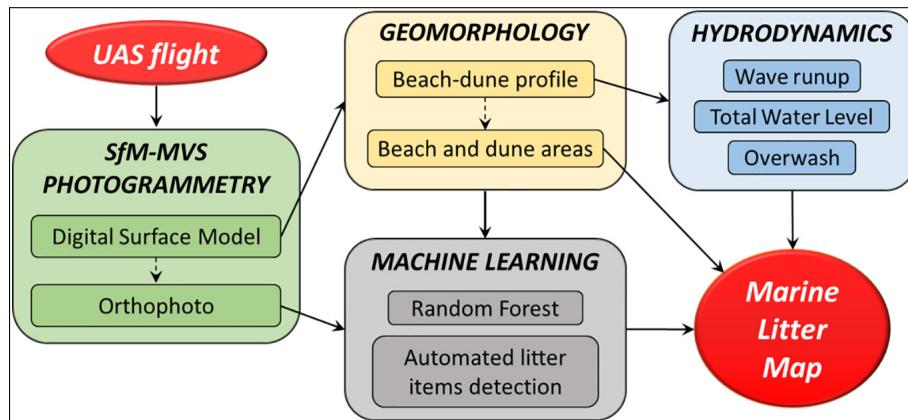


2.5. ábra. A tengeri hulladékdetectáláshoz való tanító adatok előállítása [16].

Wolf és tsai. tengerparti hulladékdetectálással foglalkoztak, úgy úszó hulladékkal, mint partra kimosott hulladékkal [17]. A kutatásban bemutatnak egy APLASTIC-Q konvolúciós neuronhálót, mely 5 pixel per cm felbontással rendelkező felvételeket klasszifikált Kambodzsa területén. A modell két fő komponensből állt: egy műanyag-hulladék detectálóból (PLD-CNN) és egy műanyag-hulladék osztályozóból (PLQ-CNN). A PLD-CNN megkülönböztette a vizet, homokot, növényzetet és műanyag alapú hulladékot 83% átlagos pontossággal, illetve megállapította, hogy az adott terület mennyire hulladékos. PLQ-CNN megkülönböztette a különböző hulladék típusokat átlagosan 71%-os pontossággal: ilyenek például az üvegpalackok, cipők, textil anyagok stb. A PLD-CNN betanításához 5515 darab  $100 \times 100 \times 3$  pixeles csempét használtak, míg a PLQ-CNN betanításához 4828 darab  $50 \times 50 \times 3$  pixeles csempékre vágták a felvételeket.

Gonçalves és tsai. gépi tanulást, geomorfológiát, fotogammetriát és hidrodinamikai modellezést kombinálnak össze azzal a céllal, hogy homokos partok környékén, illetve homokdombokon detektáljanak hulladékot drónfelvételek segítségével (2.6 ábra) [18]. A geomorfológiai vizsgálat segítségével megtudták különböztetni a partokat és a homok-

dombokat, mellyel optimizálni tudták a hulladékdetectálást. A gépi tanulási modell egy Random Forest volt, mellyel a homokban található hulladékot detektálták. A hidrodinamikai számolások segítségével meg lehetett becsülni, hogy hol kerülnek a hulladékok kimosásra, illetve a parton található hulladékok mennyit fognak ott maradni. Összességében ezzel a módszerrel a hulladékdetectálás 75%-os F-Score-al sikerült.

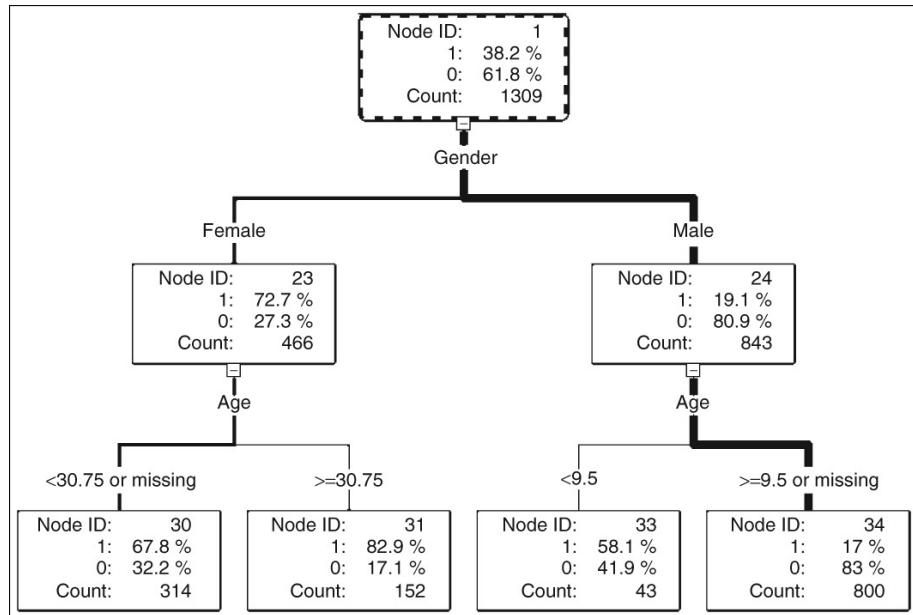


2.6. ábra. A gépi tanulás, geomorfológia, fotogammetria és hidrodinamikai modellezés használása hulladékdetectálásra [18].

Youme és tsai. konvolúciós neuronhálót (CNN) használtak, hogy a Szenegál folyó mentén hulladékot detektáljanak Saint-Louis-ban [19]. A kutatáshoz magas felbontású drónfelvételeket használtak. Drónfelvételeket készítettek 5, 10, 30, illetve 300 méter magasságon, úgy, hogy a látási szög merőleges volt a földre. A 10 méter és 30 méteres felvételeket használták a modell betanítására. A kiválasztott kutatási területnek az az előnye, hogy változatos környezetben található, így modell betanítására és tesztelésére is alkalmas. A modell átlagos pontosságát nem mérték, mivel nagy eltérések voltak a vizsgált régiók függvényében. A cikkben említi, hogy a modell pontossága lényegesen növelhető, ha több adattal kerül betanításra.

## 2.2. Döntési fa

A döntési fák általános célú predikcióra és osztályozásra használt eszközök [20]. Az előnyei közé tartozik az, hogy tekintve az egyszerű struktúrájukat, könnyen értelmezhetőek és rugalmaskak. A 2.7 ábrából látható a döntési fa működési elve.



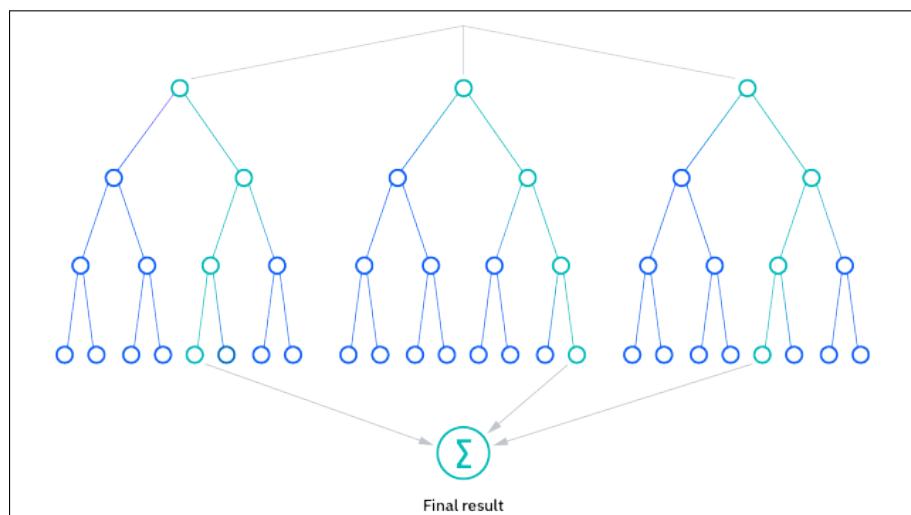
2.7. ábra. A döntési fa működési elve [20].

A 2.7 ábrán látható döntési fa a Titanic utasairól készített túlélési kutatás adatai alapján épült [21]. A fa csúcsain található 1-es címke képviseli a túlélők arányát, míg a 0-ás címke az elhunytak arányát. Mivel a 0 és 1 osztályok közül választhat a döntési fa, ezért ez egy osztályozó döntési fa. Ezen felül a csúcsban el van tárolva az is, hogy hány mérést tartalmaz. A gyökérkcsúcs tartalmazza az összes mérést. Ahhoz, hogy leágazzunk egy csúcsból, meg kell határoznunk a közvetlen gyerekcsúcsok "bemeneteit": ezek olyan mezők az adathalmazban, melyek a legjobban leírják a változékonyságát az adatoknak az adott szinten. A legalkalmasabb bemenetek meghatározása egy nyitott kutatási téma. Ebben a példában a gyökérkcsúcs közvetlen gyerekeinek a bemenete az utas neme. A következő szint bemenete pedig az utas kora. Így például a következő információkat olvashatjuk le a döntési fáról: a 30.75 év fölötti nőknek volt a legnagyobb túlélési esélyük, míg a 9.5 fölötti vagy ismeretlen életkorral rendelkező férfiaknak volt a legkisebb túlélési esélyük. Így tehát ha meg szeretnénk állapítani, hogy mennyi lett volna a túlélési esélye egy 35 éves férfinek, akkor látnánk azt a döntési fa alapján, hogy ez 17% lenne.

## 2.3. Random Forest

A 2.2 fejezetben tárgyalt döntési fa egyik hátránya az, hogy hajlamos a túltanulásra. A Random Forest egy felügyelt gépi tanulási módszer, mely magas pontossággal tud rátanulni a tanító adatokra (túltanulás nélkül), és jól kezeli a zajt [4]. A módszer több, véletlenszerűen paraméterezett döntési fa felépítéséből kapta a nevét: miután felépítettük ezt a

"döntési fa erdő", az adatokat úgy lehet osztályozni, hogy egy többségi szavazást hajtunk végre az összes fa eredménye szerint (2.8 ábra). Ennek a megközelítésnek köszönhetően az egyes túltanult döntési fák kiegyenlítik egymást, így a teljes erdő nem lesz túltanítva. A fák felépítéséhez több stratégia létezik, és ezen stratégiák alapján lehet finomhangolni a modell pontosságát és méretét. Az utóbbit fontos szem előtt tartani, tekintve arra, hogy elég sok tanító adat esetén a modell mérete lényegesen megnőhet helyes paraméterezés hiányában. Az ilyen paraméterek például a fák maximum mélysége, a fáknak átadott részadathalmaz dimenziói, egy csúcs kettéválasztásának a kritériumjai, a tanító adatok súlyai stb. A kutatásomban ezt a modellt tanítom be, illetve paraméterezem azzal a céllal, hogy megbízható klasszifikációt tudjon biztosítani. A modell minden képkockát osztályozni fog, így a bemeneti adatok az adott terület spektrális sávjai, illetve indexei.



2.8. ábra. A Random Forest működési elve [22].

## 2.4. PlanetScope

A Planet Labs rendszeresen műholdakat küld az űrbe azzal a céllal, hogy minden nap készüljön műholdfelvétel a Földről [23]. A projekt jelenlegi állapotában már ott tartanak, hogy majdnem napi szintű felvételeket készítenek a Föld felszínéről több, mint 430 Dove és SuperDove műhold segítségével [24]. A legújabb műholdcsaládjuk a SuperDove műholdakból áll. Ezek a műholdak a PSB.SD nevű műszert használják felvételek készítésére. A műszer a 47 megapixeles "PSBlue" szenzort használja. Ezekkel a műholdakkal 2020. március közepe óta készítenek felvételeket, melyek a vörös (red), zöld (green), kék (blue), közeli infravörös (near infrared vagy NIR), illetve az úgynevezett "red edge", "green I", "coastal blue", és sárga (yellow) sávokat is tartalmazzák. Egy csempe körülbelül 32.5 x

19.6  $km^2$  területet fed le [25], körülbelül 3 méter per pixel felbontással. A 2.1 táblázatból látható, hogy a PSB.SD műszer egyes sávjai milyen hullámhosszal rendelkeznek. A hullámhossz mellett jelenített "fwhm" a "Full width at half maximum" értéknek a rövidítése.

Sáv azonosító	Sáv neve	Hullámhossz (fwhm)
1	Coastal Blue	443 (20)
2	Blue	490 (50)
3	Green I	531 (36)
4	Green	565 (36)
5	Yellow	610 (20)
6	Red	665 (31)
7	Red Edge	705 (15)
8	NIR	865 (40)

2.1. táblázat. A PSB.SD műszer hullámhosszai [25].

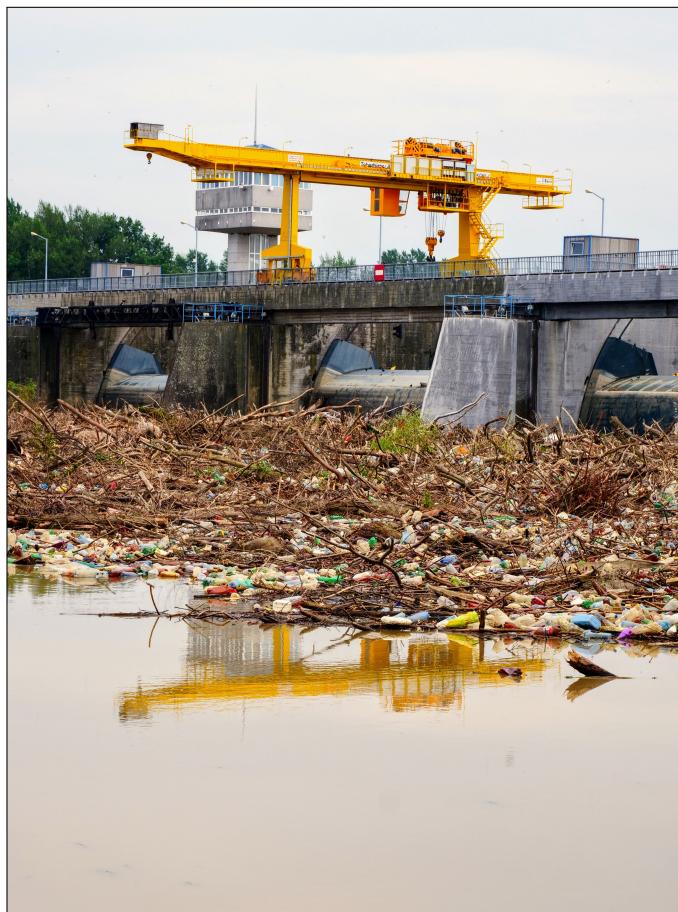
## **3. fejezet**

# **Betanítás**

### **3.1. Tanító adatok**

A betanításhoz 29 romániai hulladéklerakó és közvetlen környezete került a tanító adatok közé, illetve a Kiskörei víztároló is. A víztároló alkalmas úszó hulladékszigetet tartalmazó tanítóadatok előállítására, tekintve arra, hogy a felgyült faágak között nagy koncentrációban jelenik meg műanyag-alapú hulladék. A 3.1 ábrán látható, hogy miként gyűl össze a hulladék a víztárolónál. A romániai hulladéklerakókat egy helyi weboldalon lehet megtalálni, a hozzájuk tartozó koordinátákkal együtt [26]. Az ott bemutatott 46 hulladéklerakó közül 29 volt alkalmas tanításra: sok hulladéklerakó be lett tömve, vagy föld alatt működik. minden hulladéklerakóhoz letöltöttem egy-egy nyári + tavaszi (továbbiakban nyári), illetve téli + őszi (továbbiakban téli) multispektrális műholdképet, melyeket kézzel annotáltam. A nyári és tavaszi képeket azért vontam egybe, mivel ezek hulladékdetectálás szempontjából hasonló adatokat eredményeztek. A tanító adatok pixelenként vannak előállítva, így a végső adathalmaz 27 millió tanító adatból (pixelből) áll. minden pixelhez hozzá van rendelve a vörös, kék, zöld, közeli infravörös sáv, illetve a *PI*, *NDWI*, *NDVI*, *RNDVI*, *SR* indexek. Az indexek kiszámolási módját a 3.2 fejezetben részletezem. Ezen felül minden pixel címkézve van a 3.1 táblázatban leírtak szerint. Nagyon fontos, hogy minél pontosabban meg lehessen állapítani, hogy a hulladéklerakókon mely területeken található hulladék, hiszen sok hulladéklerakón törmeléket is tárolnak, ezt általában egy külön területen. Emiatt a tanító adatokat a következő módszerrel állítottam elő: Megvizsgáltam Google Maps segítségével [27], hogy az adott hulladéklerakónál hol tárolnak törmeléket és hol tárolnak műanyag alapú hulladékot. A Google Maps légi felvételei elég magas felbontással rendelkeznek ahhoz, hogy általában szemmel meglehessen

különböztetni a műanyag alapú hulladékot a törmeléktől. A 3.2 ábrából látható, hogy még a törmelék inkább fehér színt tartalmaz, addig a műanyag alapú hulladék párásabb, tekintve arra, hogy a műanyag sokszor színezve van. Emellett, a hulladéklerakókat nagyon konzervatívan jelöltem ki: csak akkor jelöltem be egy területet, mint hulladékos terület, ha a felvételből egyértelmű volt, hogy az adott terület műanyag alapú hulladékkal szennyezett. Természetesen, helyi terepvizsgálattal, illetve magasabb felbontású felvételek készítésével pontosabb adatokat lehetne előállítani.



3.1. ábra. A kiskörei víztároló hulladékторlasza [28].

Címke azonosítója és színe	Címke neve	Címke magyarázat
100	Hulladék	Azon területek, melyeken hulladék van.
200	Víz	Olyan területek, melyeken kizárolag víz van, általában folyók.
300	Legelők/Erdők	Zöld övezetből álló vad területek. Ezek lehetnek fák lombjai vagy füves zónák.
400	Mezők	Olyan földes területek, melyek meg vannak művelve, illetve ahol mezőgazdasági növények találhatóak, például gabonafélék.
500	Ismeretlen	Olyan területek, melyek a korábbi kategóriákba nem sorolhatók bele. Ilyenek az épületek, aszfaltozott utak, háztetők, mezei utak.

3.1. táblázat. A tanító adatok címkéi.



(a) Műanyag alapú hulladék.



(b) Törmelék.

3.2. ábra. A műanyag alapú hulladék, törmelék mellé helyezve. Forrás: Google Maps.

## 3.2. Használt sávok és indexek

A kutatás során felhasználom a PlanetScope műholdak kék (Blue), zöld (Green), Vörös (red), közeli infravörös (Near-Infrared vagy NIR) sávjait, illetve a kutatólaborban már számolt indexeket. Pontosabban a Plastic Index (3.1 képlet), Normalized Difference Water Index (3.2 képlet), Normalized Difference Vegetation Index (3.3 képlet), Reversed Normalized Difference Vegetation Index (3.4 képlet), Simple Ratio (3.5 képlet) indexek kerülnek számolásra [29, 5].

$$\text{Plastic Index (PI)} = \frac{\text{NIR}}{\text{NIR} + \text{Red}} \quad (3.1)$$

$$\text{Normalized Difference Water Index (NDWI)} = \frac{\text{Green} - \text{NIR}}{\text{Green} + \text{NIR}} \quad (3.2)$$

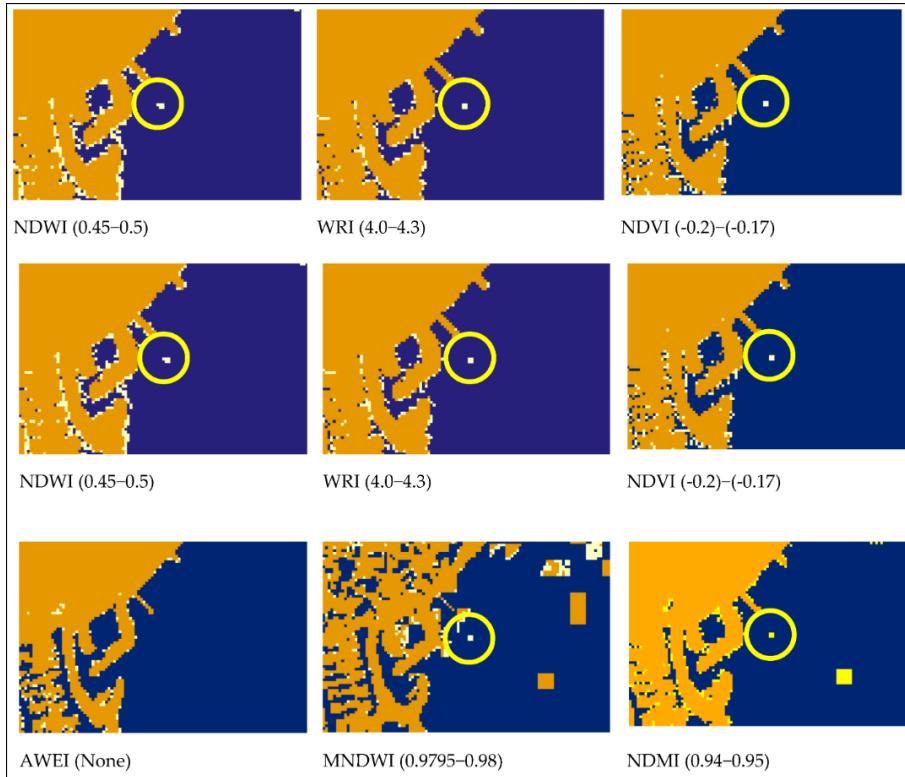
$$\text{Normalized Difference Vegetation Index (NDVI)} = \frac{\text{NIR} - \text{Red}}{\text{NIR} + \text{Red}} \quad (3.3)$$

$$\text{Reversed Normalized Difference Vegetation Index (RNDVI)} = \frac{\text{Red} - \text{NIR}}{\text{Red} + \text{NIR}} \quad (3.4)$$

$$\text{Simple Ratio (SR)} = \frac{\text{NIR}}{\text{Red}} \quad (3.5)$$

A Plastic Index, ahogy a neve is sugallja, egy olyan index, mely a vízen úszó hulladékok pixelein magas értékeket, míg a vízpixeleken alacsony értékeket vesz fel. A számításban kihasználásra kerül az, hogy a közeli-infravörös tartományban a hulladék sokkal jobban visszaverődik a vízhez képest, mint a vörös tartományban. Az NDWI a NIR és zöld sávokat használja fel a vízben található tárgyak, illetve növények kijelölésére [30]. Az NDVI -1 és +1 értékek között tartózkodik. Ha adott területen az NDVI negatív értéket vesz fel, akkor nagy az esélyünk arra, hogy a területet víz borítja. Ha 1-hez közelí értéket vesz fel, akkor nagy eséllyel növényzet borítja az adott területet. Az RNDVI ennek a -1-szerese. A Simple Ratio is a vörös és közeli infravörös tartományokat használja ki a vegetáció további detektálásához és osztályozásához.

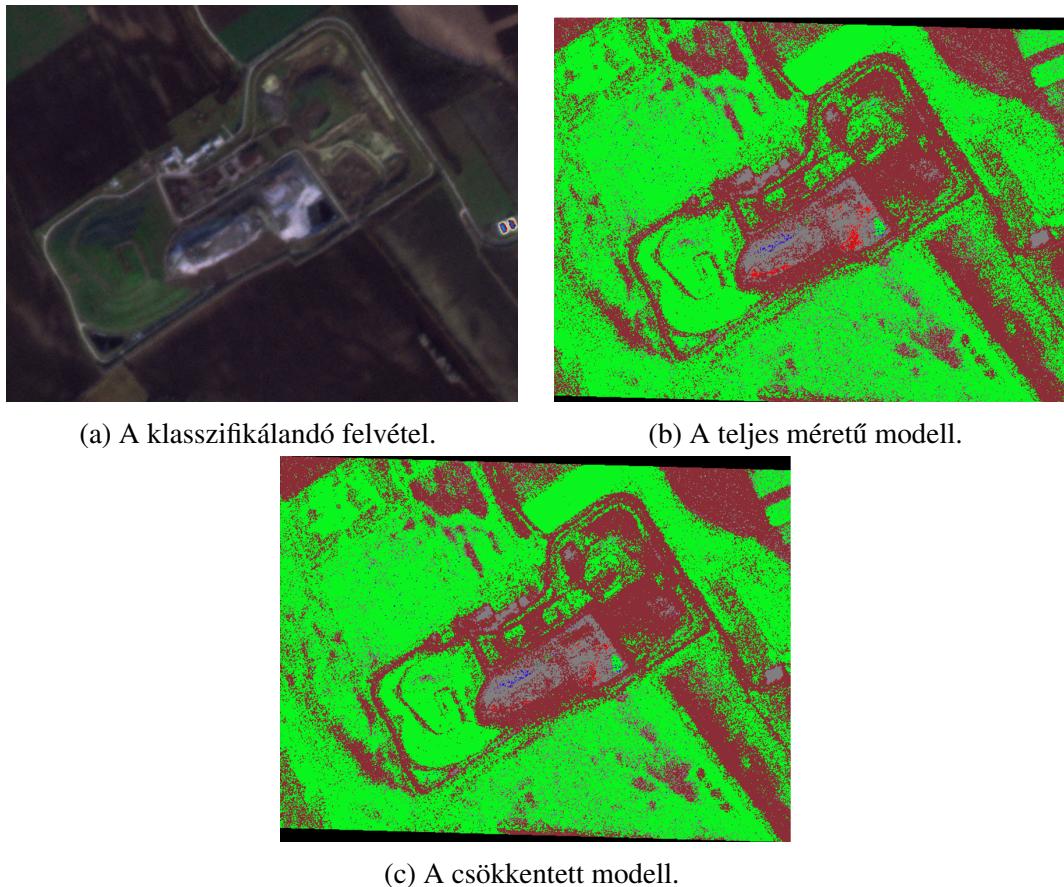
Themistocleous és tsai. összehasonlították ezeket az indexeket (3.3 ábra): Összekötöttök több műanyag palackot, és felhelyezték a vízre. Ezután Sentinel-2 felvételekkel összehasonlították a különböző indexek teljesítményét ennek a szigetnek a megtalálásában. A kutatás eredménye az volt, hogy a Plastic Index tudta a legjobban kiemelni a műanyag-szigetet [29].



3.3. ábra. Különböző indexek összehasonlítása egy mesterségesen előállított úszó műanyag-palack szigeten [29].

### 3.3. Tanítási paraméterek

A Random Forest betanításához a Scikit-Learn Python csomagot használtam [31]. A nagy adathalmaz miatt a Random Forest modell is nagyon nagy lett (körülbelül 14GB), ami egy nehezen kezelhető méret, így érdemes módosítani a modell paraméterein, hogy ez kisebb méretű legyen. A legjobb eredményeket az azzal értem el, hogy a Random Forest fák méretét 20 mélységre limitáltam. Ennek köszönhetően a model méretét 2GB-ra tudtam csökkenteni, és a 3.4 ábrából látható, hogy a csökkentett modellben enyhén megnő a hulladékra vonatkozó *false negative* ráta, míg a *false positive* arány nem nő, de cserében egy kezelhető méretű modellt kapok.



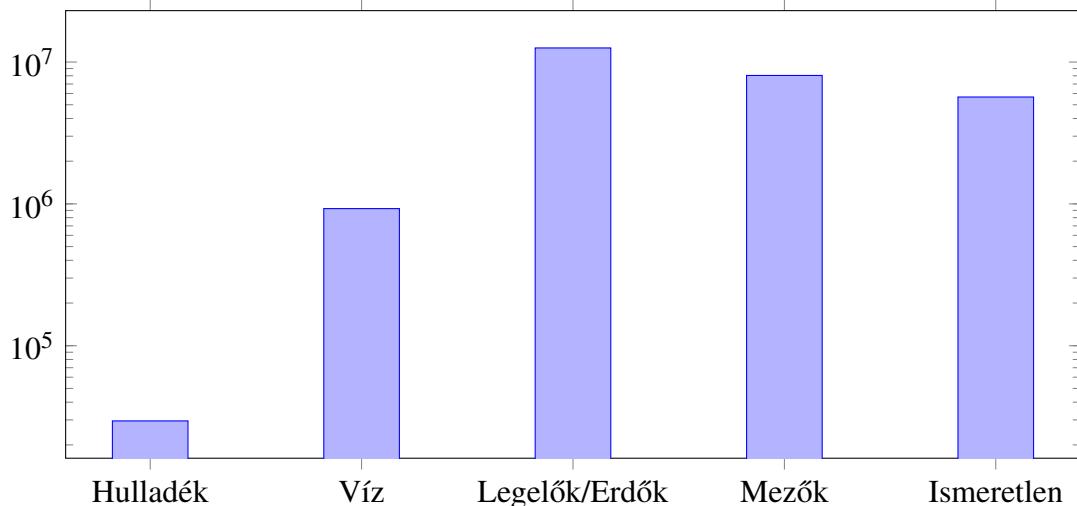
3.4. ábra. A csökkentett modell keveseb *true positive* értéket termel a teljes méretű modellhez képest a pusztazámi hulladéklerakó egyik felvételén. Felvétel dátuma: 2021.12.08.

Továbbiakban felmerült az a probléma is, hogy a tanító adatok nagyon aránytalanok: A 3.5 ábrából látható, hogy nagyságrendekkel kevesebb adattal rendelkeztünk hulladékról, mint az összes többi adatról. Emiatt a modell nagyon sok false-negatívet termelt. Ennek korrigálására súlyokat alkalmaztam a tanító adatokra. A súlyok kiszámolásához az összes címkére a 3.6 képletet használtam.

$$\text{címke súlya} = \frac{\text{adathalmaz mérete}}{\text{címke darabszáma}} \quad (3.6)$$

### 3.4. Felvételek normalizálása

Az egyik gyakori probléma gépi tanulásban az, hogy ha elég nagy eltérések vannak a felvételek között, például időjárás miatt, akkor a modell hajlandó félreosztályozni egyes területeket. Illetve, a műholdak, amikkel készülnek a felvételek idővel újabb műszereket kapnak, amik eltéréseket termelhetnek a felvételeken. Ennek fényében időben is lehet-

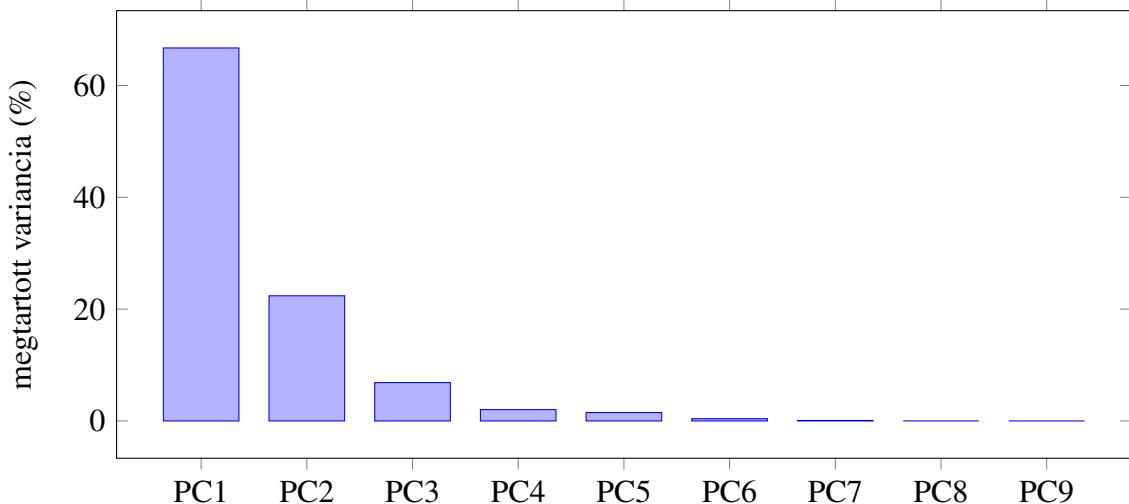


3.5. ábra. Az adatok közötti aránytalanság, logaritmikus skálázással.

nek lényeges eltérések a felvételek között. Ennek korrigálására érdemes megvizsgálni a műholdfelvételek normalizálását. A normalizálás egy referencia kép szerint történik: kiválasztok egy referencia felvételt egy adott területről és időszakról (nyár, télap), és az összes többi felvételt arról a területről és időszakról erre a felvételre normalizálom. Így a nagymértékű eltérések csökkentve lesznek a modell számára, és várhatóan jobban fogja osztályozni a felvételeket, amik különböző körülmények között voltak előállítva. A normalizáló algoritmust az ELTE IK térinformatikai laboron belül fejlesztette az egyik kollégám. A normalizált felvételekhez külön kell egy modellt betanítani. Így a teszthalmazban minden területhez és minden évszakhoz kiválasztottam egy-egy felvételt, mint referencia kép. Ezután minden műholdképet normalizáltam a referenciához. A módszer eredményeit a 4.6 fejezetben részletezem.

### 3.5. Főkomponens analízis (PCA)

A modell méretének a csökkentésére megvizsgáltam a főkomponens analízis (PCA) alkalmazását is [32]. A főkomponens analízis a gépi tanulásban egy szélesen elterjedt módszer. A módszer lényege az, hogy egy többdimenziós adathalmazból kivonja a legfontosabb információkat egy alacsonyabb dimenziószámú adathalmazba. Ezeket nevezzük főkomponenseknek. A főkomponensek korrelálatlanok, illetve a megfigyelési egységek varianciájának a nagy részét az első pár főkomponensben tároljuk [33]. Azt, hogy hány főkomponensem szeretnénk megtartani, akár empirikus módon is meghatározhatjuk annak függvényében, hogy mekkora mértékben szeretnénk megtartani az eredeti adat-



3.6. ábra. A főkomponensek varianciája a tanító halmazon. 90% variancia megtartásának érdekében elég az első három főkomponenst kiválasztani.

halmaz varianciáját. Automatizálás érdekében használható az Elbow-method [34]. A 3.6 ábrán látható, hogy ennek az adathalmaznak az esetében ha 90%-át szeretném megtartani a varianciának, akkor elég az első három főkomponenst megválasztanom. Így a továbbiakban, amikor a PCA alkalmazására hivatkozok, akkor az első három főkomponens kiválasztására gondolok.

A PCA használatának a motivációja az volt, hogy a bemeneti adatok dimenziószámának a redukálásával csökkenni fog a modell mérete, de érdekes módon a modell mérete nem csökkent, sőt elég kevés főkomponens megválasztásával lényegesen megnőtt. Ezt az is tükrözi, hogy megnőtt átlagosan a Random Forest döntési fáinak a mérete, minél kevesebb dimenziószámú adatot kapott. A 3.2 táblázatból látható, hogy különböző főkomponenseknél mekkora volt átlagban a fák mérete az adatok dimenziószámának függvényében. További vizsgálatok után kiderült, hogy hogyha kevesebb dimenziójú adatot adtam a modellnek, akkor a mérete lényegesen megnőtt.

Ezen felül az is célja volt a PCA alkalmazásának, hogy a kiszámolt indexek információt megtartva, alacsonyabb dimenziószám segítségével a Random Forest modell hatékonyabban fogja majd feldolgozni a műholdfelvételeket. [35] megmutatja, hogy egyes modellekben jobb klasszifikációt tudtak elérni több spektrális sávból álló adat esetén. Ennek az az oka, hogy ha igen sok a korreláció a különböző dimenziók között, akkor a modell rátanulhat a zajra. Mivel a modell által használt indexek között sok a korreláció, érdekes megvizsgálni a PCA-t olyan szempontból is, hogy esetleg javít-e a klasszifikációs eredményeken.

Főkomponensek száma	Fák méretének mediánja
Főkomponensek nélkül (9 dimenzió)	71.5
5 főkomponens	71
4 főkomponens	73
3 főkomponens	85

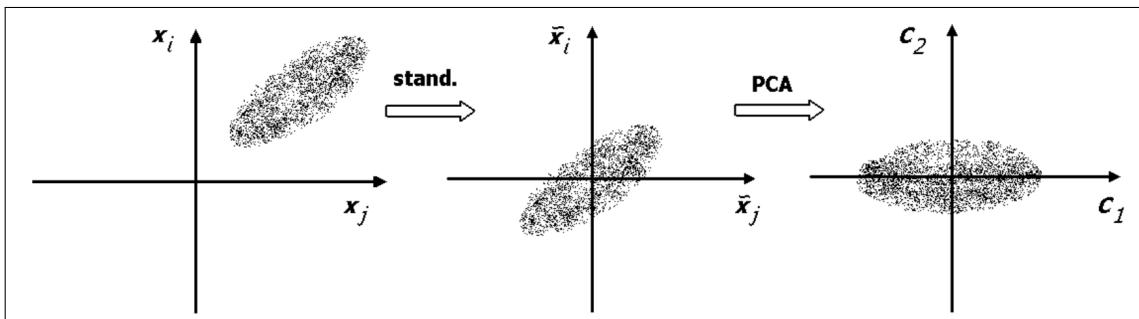
3.2. táblázat. A döntési fák méretének a mediánja nem csökkent, amint a főkomponensek száma csökkent, cserében 3 főkomponensnél már nőtt.

A PCA alkalmazása a Random Forestre a következő lépésekkel áll:

1. A tanító adatok standard skálázása a 3.7 képlet szerint.
2. A főkomponens analízis alkalmazása a tanító adatokra.
3. A Random Forest modell betanítása a tanító adatokon.

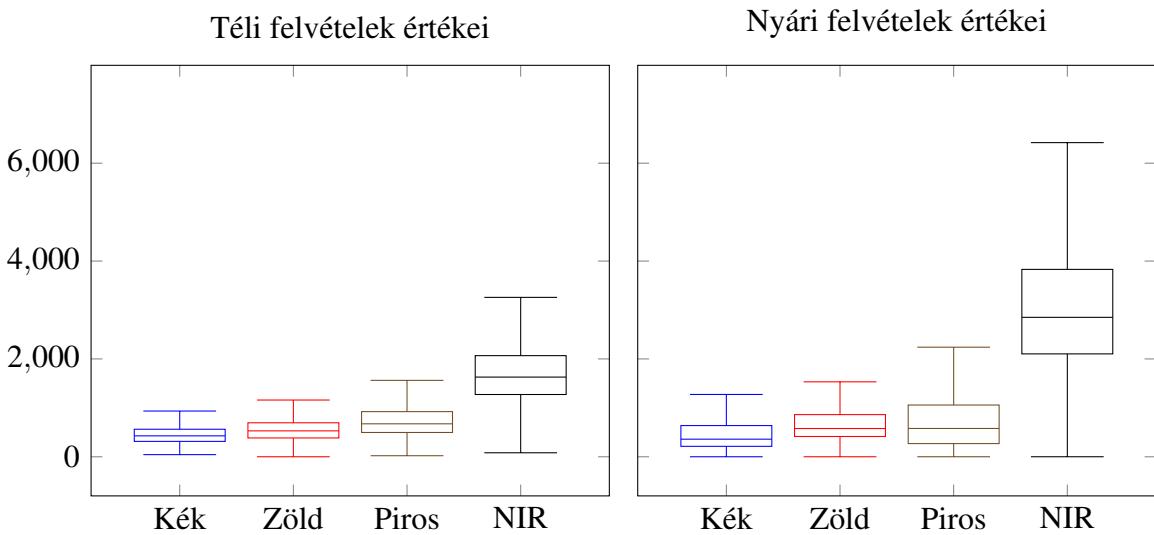
A főkomponens analízis folyamatának a geometriai jelentését a 3.7 ábra illusztrálja.

$$f(x) = \frac{x - \text{Average}}{\text{Standard Deviation}} \quad (3.7)$$



3.7. ábra. A főkomponens analízis geometriai jelentése: a standardizálás 0 várható értékűvé és 1 empirikus szórásúvá teszi a változókat, vagyis a pontfelhőt betolja az origóba, majd elforgatja a legnagyobb variancia irányába, ami az első főkomponens [33].

A modell tesztelésére is ugyanezeket a lépéseket kell elvégezni. A főkomponens analízis használásához a Scikit-Learn [31] Python programcsomagot használtam. A főkomponens analízissel betanított modellt a 3.3 fejezetben leírtak szerint paramétereztem. A 4.3 fejezetben részletezem a főkomponens analízissel tanított modell teljesítményét.



3.8. ábra. Nyári és téli adatok összehasonlítása.

### 3.6. Nyári és téli adatokra való lebontás

Alapértelmezetten a nyári és téli adatok között lényeges különbség tud lenni távérzékelés szempontból Közép-Európa területén: a téli időszakokban gyérebb a vegetáció, ködösebb a levegő, illetve a nap sem süt ugyanabból a szögből. Ez befolyásolhatja a modell pontosságát is az adott időszakban. A nyári időszakot márciustól októberig tartó időszakként definiáltam, és a téli időszak pedig novembertől februárig tart. Az időszakok aszerint vannak megválasztva, hogy mikor leveleznek ki, illetve hullatják ki a leveleiket a fák. Valóban, az októberi időszakban már inkább sárgásak lesznek a levelek, de az októberi tanító halmaz mérete önmagában igen kicsi érdemi tanításra. A 3.8 ábrából látható, hogy főleg a közeli infravörös (NIR) sávokon nagy eltérések vannak a nyári és téli felvételek között. Ennek fényében betanítottam külön egy nyári és egy téli modellt. A nyári modell teljesítményét a 4.5 fejezetben részletezem a tesztadathalmaz segítségével, míg a téli modell teljesítményét a fejezetben mutatom meg, empirikusan.

## 4. fejezet

### Verifikáció

#### 4.1. A teszthalmaz

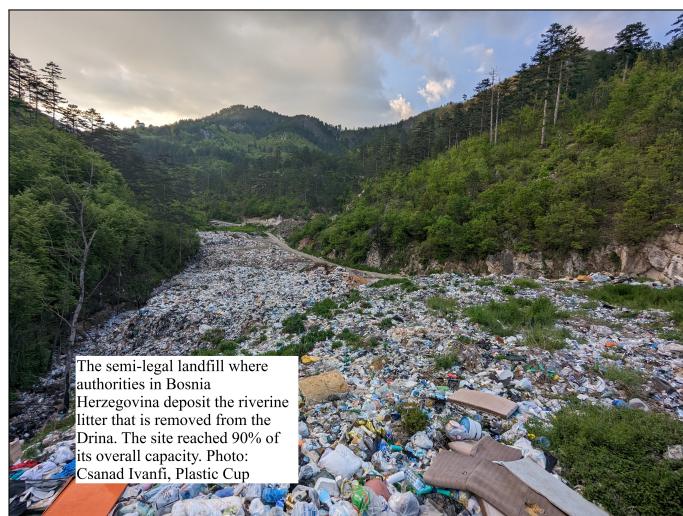
A teszthalmaz 3 nyári drinai felvételből áll, ahol a hulladékkal szennyezett területeket kézzel annotáltam. Ez a terület egyben egy szárazföldi hulladéklerakót (4.2 ábra), illetve egy vízfelszíni hulladékszigetet is tartalmaz, így alkalmas minden detektálásnak a tesztelésére. A 4.1 ábrából látható, hogy Drinán úgy fogják meg az úszó műanyag-alapú hulladékot, hogy egy zsinórba rálépnek üres hordókat, melyek a víz felszínén lebegnek. Így, minden, ami elég könnyű ahhoz, hogy a folyó felszínen ússzon (műanyagpalackok, kisebb fadarabok) megakad a hordók mögött, míg például nagyobb fadarabok, vagy más, nehezebb uszadékok a zsinór alatt elúsznak. Így a folyó felszínén kialakuló sziget nagy koncentrációban tartalmaz műanyag alapú hulladékot, tehát alkalmas arra, hogy a modellt ezen validáljam vízfelszíni hulladékdetektáláshoz. Ráadásul erről a területről nem készültek tanító adatok ebben a kutatásban, így a modell teljesítménye az itteni felvételeken jól tesztelhető. Ezen kívül téli drinai és nyári kiskörei felvételeken is lefuttattam a modelleköt, ezek a 4.7 fejezetben vannak vizsgálva.

A 4.3 ábrából látható egy-egy vizuális összehasonlítás a régi és az új modell klasszifikációja között a teszthalmaz egyik felvételén. A hulladékos területek pirossal vannak jelölve. Látszik ezen a példán, hogy az új modell több *false negative*-ot termel főleg a hulladéksziget körül, de ugyanakkor lényegesen lecsökkenti a *false positive*-ok arányát a régi modellhez képest. Ráadásul a folyó mellett található hulladéklerakót is megtalálja az új modell, míg a régi modell nem találja meg, illetve a régi modell a lerakó környékét és az utakat, épületeket gyakran hulladéknak detektálja. Ez egy fontos eredmény, hiszen amint a 1.2 fejezetben is tárgyaltam, célja ennek a kutatásnak, hogy csökkentsem a modell

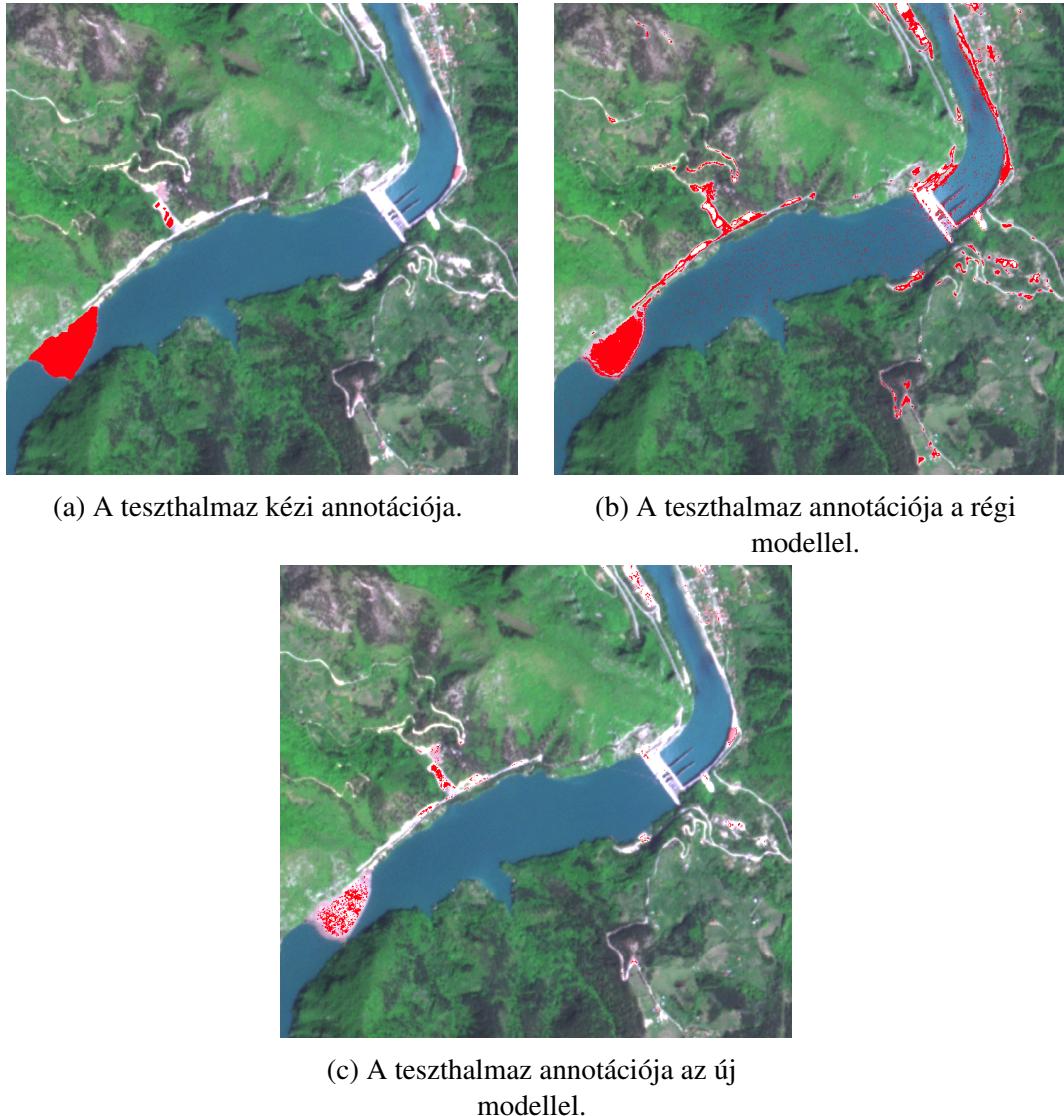
*false positive* arányait, miközben továbbra is meg tudja találni a hulladéklerakókat, illetve hulladékszigeteket.



4.1. ábra. A drinai hulladéksziget. Egy lebegő zsinór fogja meg a műanyagpalackokat [36].



4.2. ábra. A Drina melletti féllegális szemétlerakó [28].



4.3. ábra. Az új modell összehasonlítása a régi modellel az egyik Drinai teszt felvételen.  
Felvétel dátuma: 2023.05.07.

## 4.2. Teljesítmény mérése

A teszthalmaz eredményeit a *Confusion Matrix* módszerével értékeltem ki [37]. Ezután ezeket az értékeket arra használtam, hogy a "Comission rate" (4.1 képlet), "Omission rate" (4.2 képlet), "Match rate" (4.3 képlet), illetve "Extraction rate" (4.4 képlet) értékeket számítsam ki [38].

$$\text{Comission rate} = \frac{N_{com}}{N_{ref}} \quad (4.1)$$

$$\text{Omission rate} = \frac{N_{om}}{N_{ext}} \quad (4.2)$$

$$Match\ rate = \frac{N_{match}}{N_{ref}} \quad (4.3)$$

$$Extraction\ rate = \frac{N_{ext}}{N_{ref}} \quad (4.4)$$

$N_{com}$ ,  $N_{om}$ ,  $N_{match}$ ,  $N_{ext}$ ,  $N_{ref}$ , rendre a *false positive*, *false negative* (a mátrix mellékátlói), *true positive* (a mátrix főátlója), a modell által detektált pozitív, illetve a referencia adatokban található pozitív értékek. Összehasonlítottam az új modell teljesítményét a régi modell teljesítményével. A 4.1 táblázatból látható a két modell teljesítményének az átlaga a három felvételen.

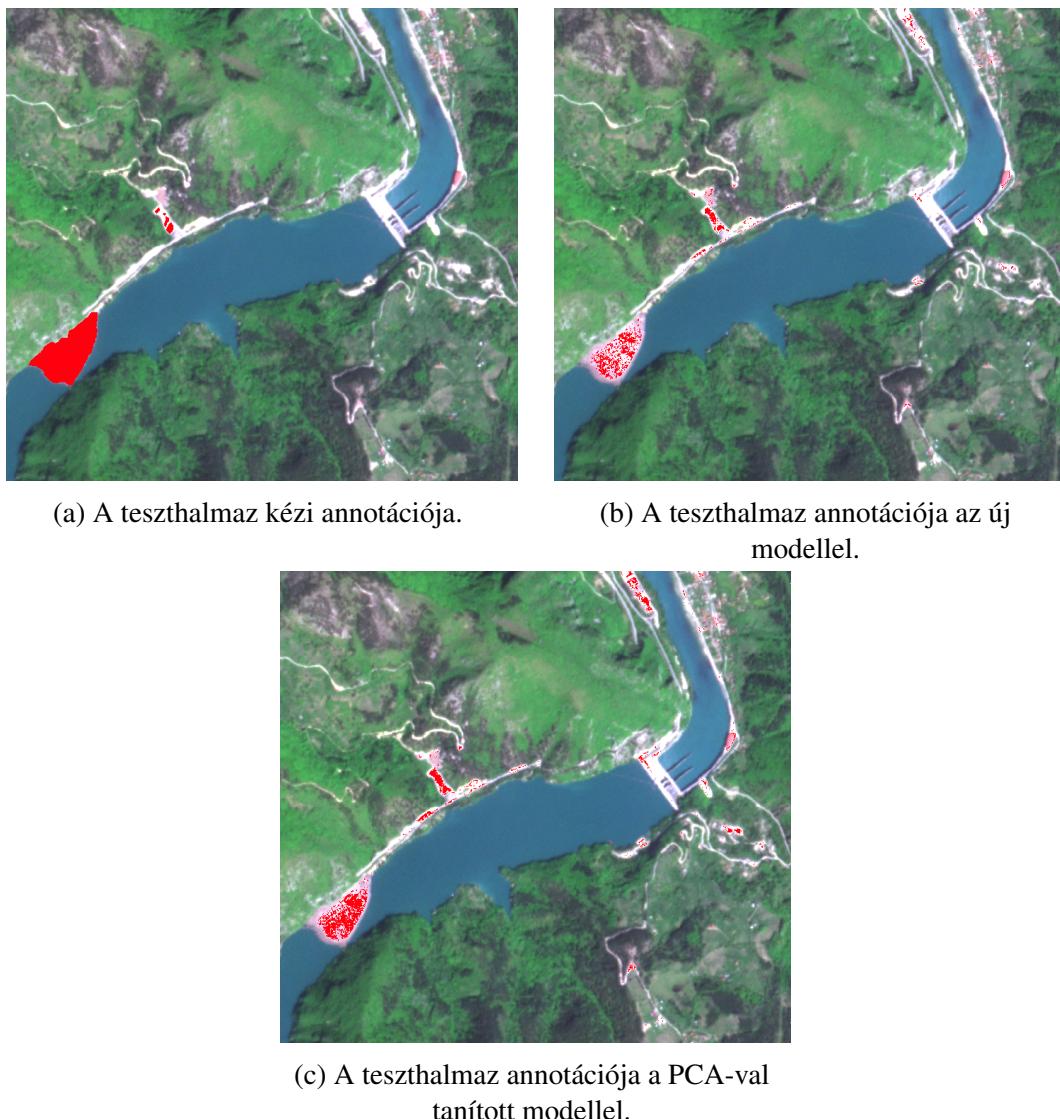
Mérés azonosító	Régi modell átlagai (%)	új modell átlagai (%)
<i>Comission Rate</i>	63.67	28.13
<i>Omission Rate</i>	26.21	70.67
<i>Match Rate</i>	73.79	29.32
<i>Extraction Rate</i>	208.18	41.31

4.1. táblázat. A régi modell és az új modell teszteredményei átlagolva.

Az új modell egy jóval kisebb *false positive* aránnyal rendelkezik mint a régi modell, de cserében a *false negative* arányok is nagyok. Ennek oka a 4.3 ábrából látható, hiszen a régi modell sokkal több pontot detektál a hulladékszigeten, míg az új modell kevesebb pontot detektál, de továbbra is nagy mértékben megtalálja a hulladékszigetet. Illetve a 4.1 fejezetben is tárgyaltam, hogy a szárazföldi hulladéklerakót a folyó mellett az új modell már megtalálja, míg a régi nem találja meg. Tekintve arra, hogy a *match rate* a *true positive*-al arányos, és az *extraction rate* az összes *positive*-al arányos, ezek az értékek is kisebbek lesznek, mint a régi modell értékei.

### 4.3. Főkomponens analízis teljesítménye

A 3.5 fejezetben részletezett főkomponens analízis módszert is összehasonlítottam az új modell teljesítményével, a *Confusion Matrix* módszerének segítségével. A 4.4 ábrából látható, hogy a főkomponens analízzssel tanított modell sokkal jobban ki tudja szűrni a vízfelszínen kialakuló zajt. Az is látható, hogy a főkomponens analízzssel kombinált Random Forest is hasonlóan tudja detektálni a hulladékkal szennyezett területeket, annyi különbséggel, hogy a hulladékos területen több pontot detektál, de cserében több *false positive*-ot termel.

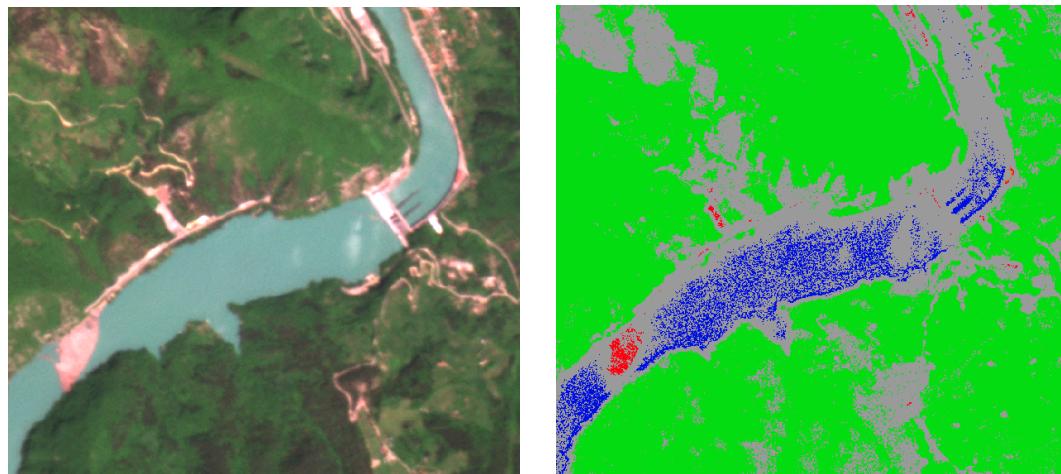


4.4. ábra. Az új modell összehasonlítása a PCA-val tanított modellel az egyik Drinai teszt felvételen. Felvétel dátuma: 2023.05.07.

A 4.2 táblázatban összesítem a főkomponens analízis mutatóit. A táblázatból leolvasható, hogy míg a főkomponens analízis picivel nagyobb *false positive* aránnyal rendelkezik, egyben kisebb *false negative* aránnyal rendelkezik. Ráadásul a felvételeken készülő zajt sokkal jobban kezeli, amint a 4.5 ábrából is látható: A főkomponens analízzssel betanított modell sokkal jobban tudta detektálni a vizet a folyón, mint a főkomponens analízis nélküli modell.

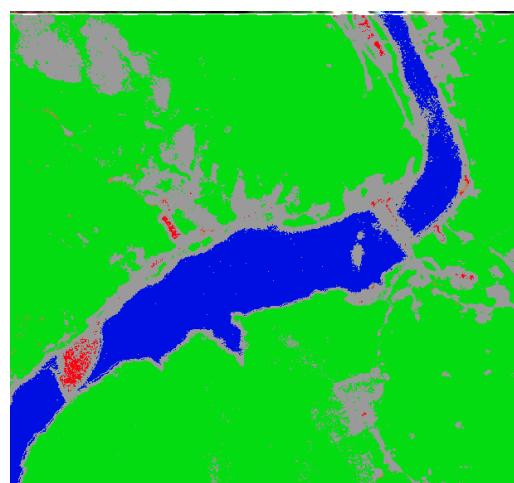
Mérés azonosító	PCA-val tanított modell átlagai (%)	PCA nélkül tanított modell átlagai (%)
<i>Comission Rate</i>	39.01	28.13
<i>Omission Rate</i>	65.00	70.67
<i>Match Rate</i>	34.99	29.32
<i>Extraction Rate</i>	65.25	41.31

4.2. táblázat. A főkomponens analízissel betanított modell teljesítményének az átlagai.



(a) Drina műholdfelvételle.

(b) A PCA nélküli modell címkézése a Drinán.

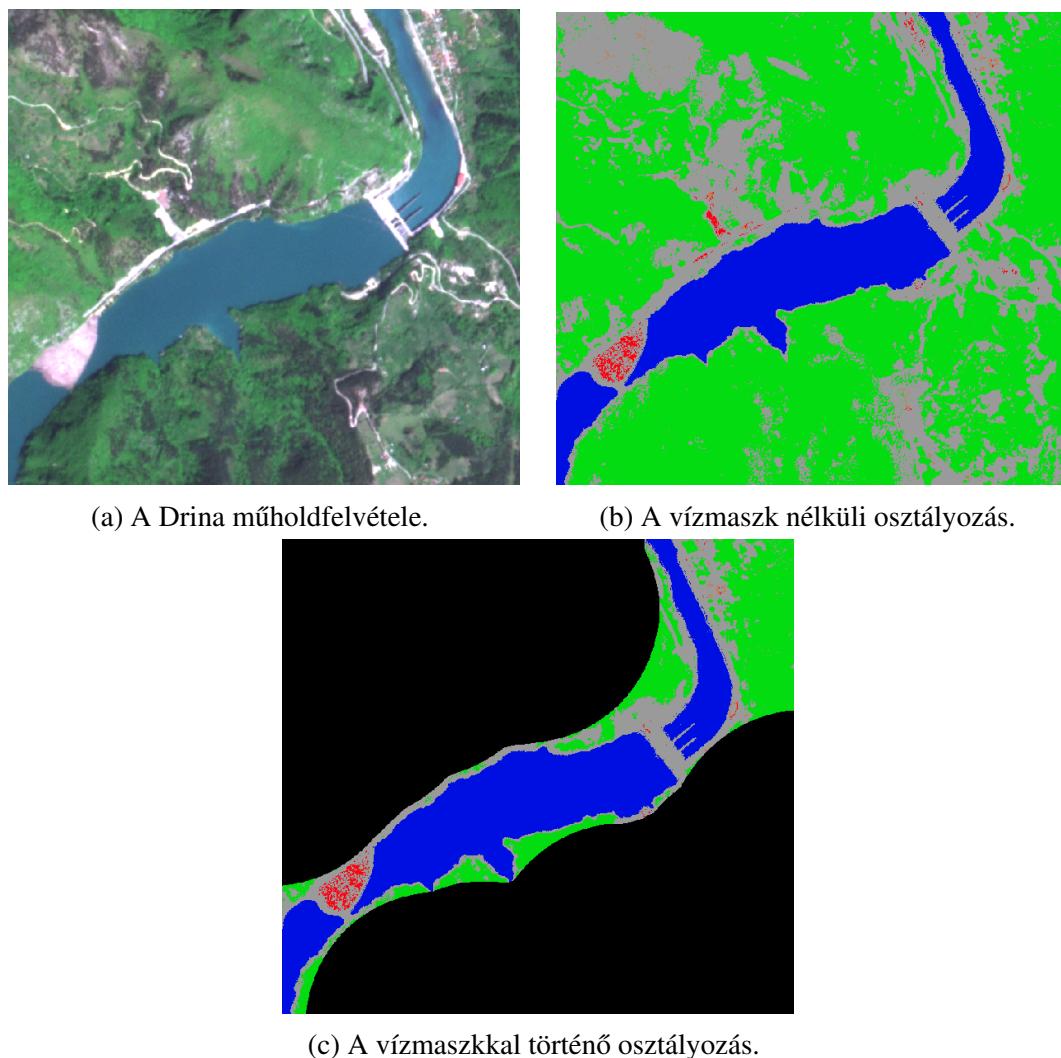


(c) A PCA-val betanított modell címkézése a Drinán.

4.5. ábra. A főkomponens analízissel betanított modell osztályozásának összehasonlítása a PCA nélkül betanított modellel. Felvétel dátuma: 2023.05.21.

## 4.4. Vízmaszkolás

A kutatásunk hosszútávú célja a folyók közelében található hulladékkal szennyezett területeknek a detektálása, így a folyóktól távolabbi területeket érdemes kivágni a fölösleges riasztások elkerülésének érdekében. Ehhez használtam egy vízmaszkolási algoritmust, amit a laboron belül elkészített az egyik kollégám, és megvizsgáltam, hogy hogyan teljesít az új modell egy vízmaszk mellett. A 4.6 ábrán látható az, hogy a modell a folyótól távoli területeket nem osztályozza, így automatikusan a folyótól igen távol levő riasztások kizárásra kerülnek. Természetesen ez paraméterezhető, így ha mégis a Drina melletti hulladéklerakót is szeretnénk vizsgálni, akkor a vízmasznak egy nagyobb távolságot lehet állítani.



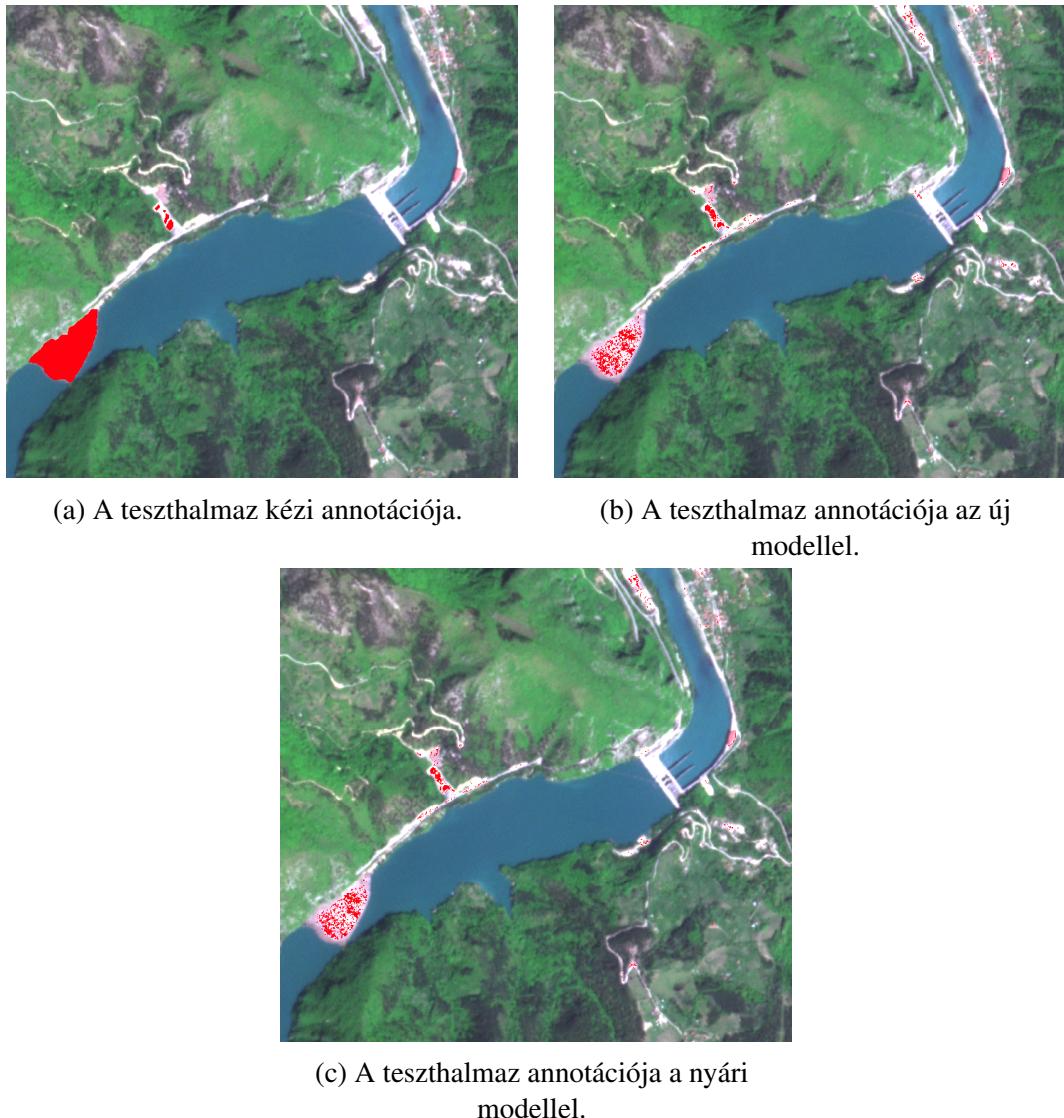
4.6. ábra. A vízmaszkolt osztályozás összehasonlítása a vízmaszk nélküli osztályozással. A Drina melletti hulladéklerakó kizárásra kerül. Felvétel dátuma: 2023.05.07.

## 4.5. Nyári adatokon betanított modell teljesítménye

A nyári adatokon tanított modellnél javulásra lehet számítani az eredményekben, tekintve arra, hogy ez a modell erre az évszakra specializálódik. A 4.3 táblázatból látható, hogy a külön nyári felvételeken tanított modell picivel kevesebb false-positive-ot tartalmaz az általánosan betanított modellnél a nyári felvételeken. A 4.7 ábrából látható, hogy a nyári modell egy picivel jobban teljesít, mint az általános modell. Természetesen abba az irányba haladni, hogy külön nyári és külön téli modellt tanítunk be, azt az implikációt vonja maga után, hogy két modellt kell karbantartani egy modell helyett. Ugyanakkor a téli modell validálása külön kihívást jelent tekintve arra, hogy téli időszakban sokszor homályosak a felvételek a magasabb csapadékszint, és felhősebb viszonyok miatt, emiatt szabad szemmel nehezebb ellenőrizni a modell teljesítményét. Ennek fényében egy további lépése lehet a kutatásnak, hogy akár személyesen, akár magas felbontású drónfelvételek segítségével a téli felvételeket külön leellenőrizzük.

Mérés azonosító	Nyári modell átlagai (%)	Az összes évszakra tanított modell átlagai (%)
<i>Comission Rate</i>	26.1	28.13
<i>Omission Rate</i>	71.27	70.67
<i>Match Rate</i>	28.05	29.32
<i>Extraction Rate</i>	39.63	41.31

4.3. táblázat. A nyári időszakra tanított modell átlagai.

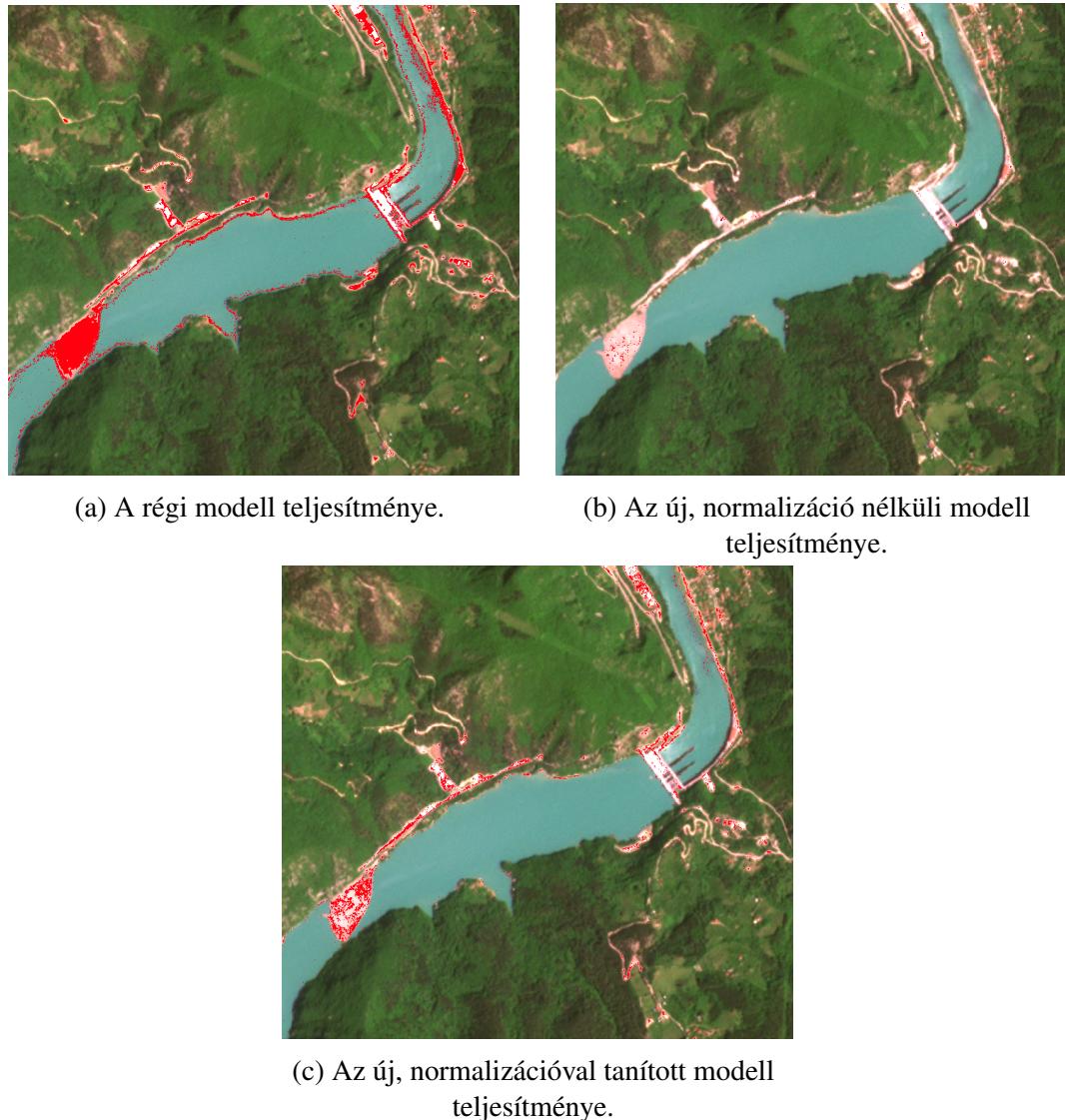


4.7. ábra. Az új modell összehasonlítása a nyári modellel az egyik Drinai teszt felvételen. Felvétel dátuma: 2023.05.07.

## 4.6. Normalizálás tesztelése

A normalizálásnak az volt a motivációja, hogy azokat a felvételeket, amik spektrális értékeikben lényeges eltéréseket tartalmaznak a többi felvételhez képest, tudjam értelmezhetővé tenni a modell számára. Teszteléshez kiválasztottam egy nyári Drinai felvételt, amin az új modell rosszul teljesített, és összehasonlítottam a normalizált képeken betanított modell eredményeivel. A 4.8 ábrán látható, hogy a régi modell ezen a felvételen detektálta a hulladékszigetet, de vele együtt detektált nagyon sok *false positive*-et is. Az új, normalizálás nélküli modell nagyon kevés hulladékot detektált, míg a normalizált modell ugyancsak sok *false positive*-et detektált. Habár első körben a normalizálással nem

a várt eredményeket értem el, egy további kutatási irány lehet ennek a továbbvizsgálása, esetleg más referenciafelvételek megválasztása.



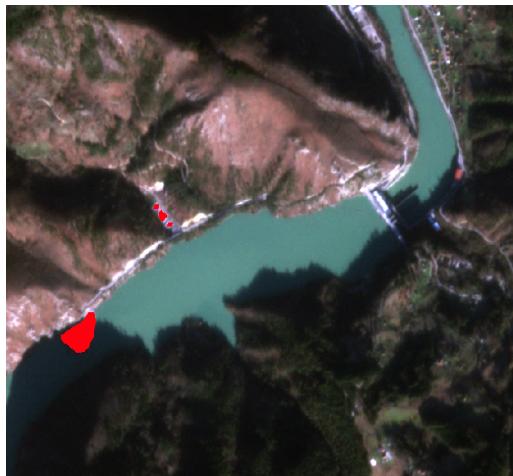
4.8. ábra. A régi, a normalizáció nélkül tanított és a normalizációval tanított modell összehasonlítása a Drinán. Felvétel dátuma: 2023.05.24.

## 4.7. Empirikus validáció

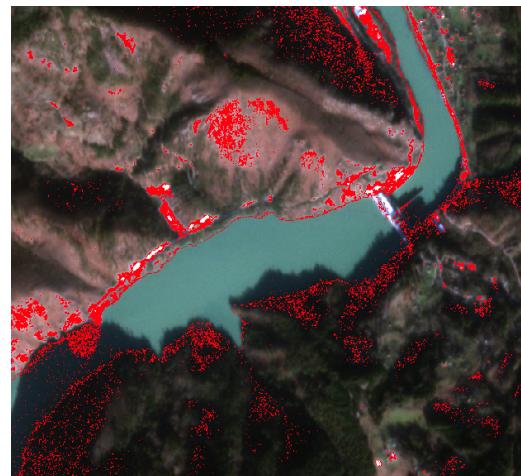
Az empirikus validáció alá esnek a téli drinai és nyári kiskörei felvételek. Ennek az az oka, hogy a téli drinai felvételek közül kihívás volt megfelelő minőségű felvételt találni numerikus validációra, míg a 2023-as kiskörei adathalmaz tanításra volt használva, így a 2024-es adathalmazból való felvételek is alkalmatlannak bizonyultak numerikus validációra.

A 4.9 ábrából látható, hogy a téli felvételen árnyék takarja a hulladéksziget felét. Itt látszik, hogy a modellek számára nehézséget jelent az árnyék alatt levő hulladéksziget detektálása. Az egyetlen modell, aki képes volt detektálni az árnyék alatti szigetet, az a régi modell volt, de cserében nagyon sok false-positive-ot termelt a többi modellhez képest. Így, a kutatás jelenlegi állapotában a téli hulladékdetectálás jelenti az egyik nagy kihívást.

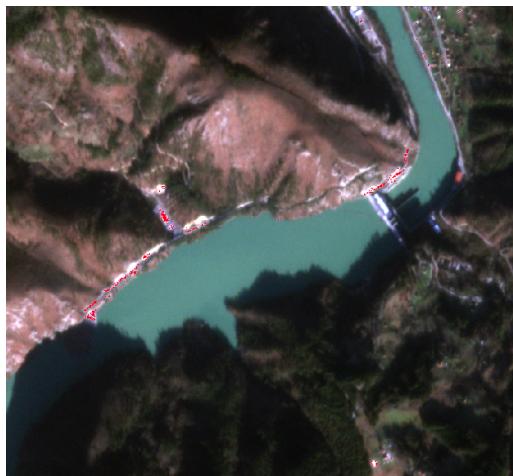
A 4.10 ábrán egy kiskörei felvételen hasonlítottam össze az összes modellt. A három legjobban teljesítő modell az új modell, a nyári adatokon betanított új modell, illetve a főkomponens analízissel betanított modell. Míg a nyári adatokon tanított modell közel teljesített az új modellhez képest, a főkomponens analízissel betanított modell több területet hulladékosnak jelölt a szeméttorlaszon, de cserében intenzívebbek is voltak a false-positive értékek.



(a) A teszthalmaz kézi annotációja.



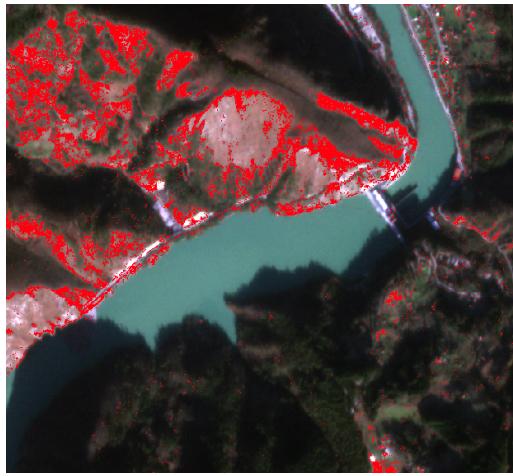
(b) A teszthalmaz annotációja a régi modellel.



(c) A teszthalmaz annotációja az új modellel.



(d) A teszthalmaz annotációja a téli modellel.

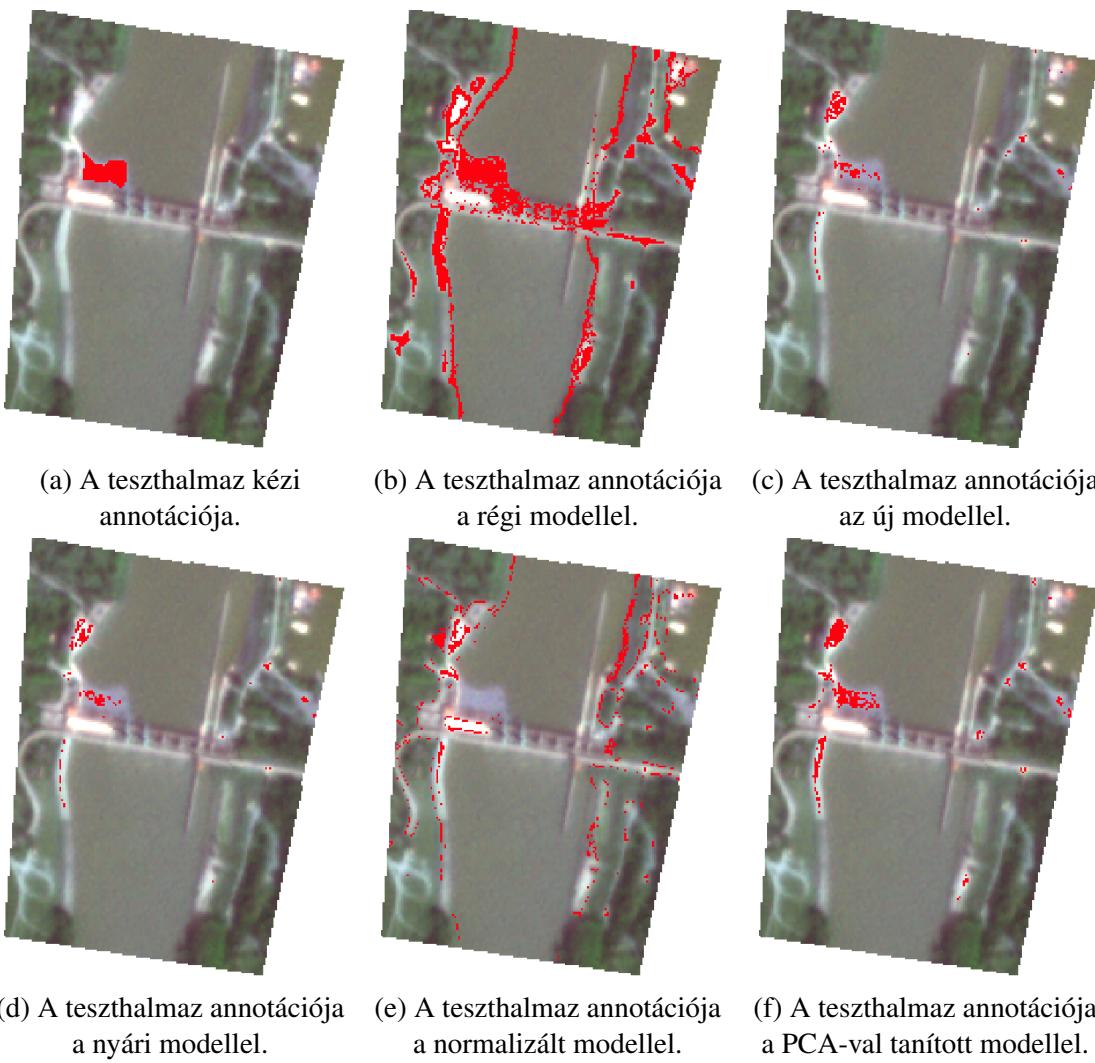


(e) A teszthalmaz annotációja a normalizált modellel.



(f) A teszthalmaz annotációja a PCA-val tanított modellel.

4.9. ábra. Az összes modell összehasonlítása az egyik téli drinai teszt felvételen. Felvétel dátuma: 2023.12.17.



4.10. ábra. Az összes modell összehasonlítása az egyik kiskörei teszt felvételen. Felvétel dátuma: 2024.04.12.

## **5. fejezet**

# **Megvalósítás és alkalmazás**

### **5.1. A meglevő alkalmazások bővítése**

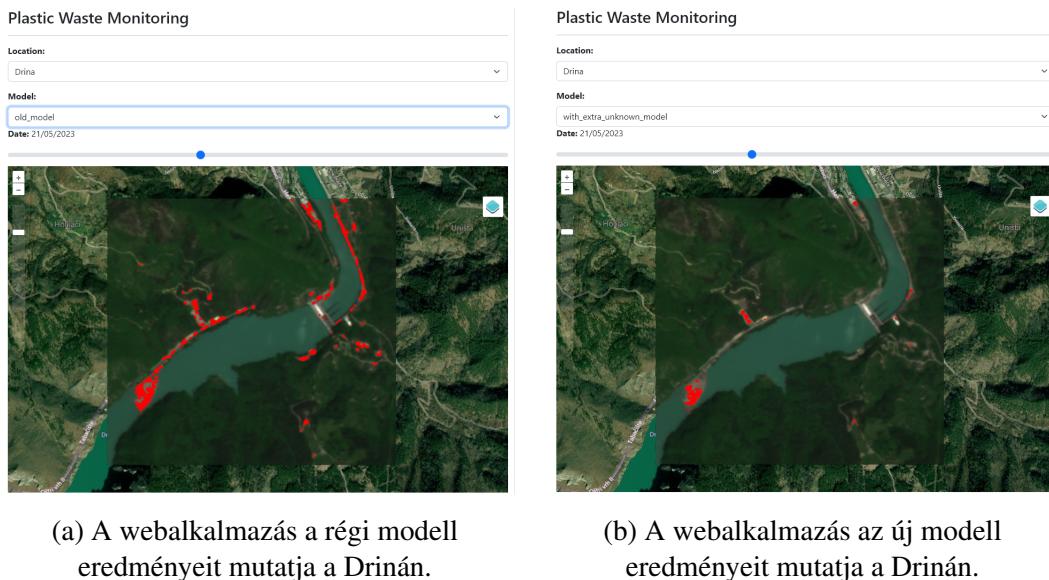
#### **5.1.1. Az asztali alkalmazás bővítése**

A meglevő asztali alkalmazás alkalmas volt a tanítóadatok hatékony előállítására, de utólag nem lehetett visszanézni, hogy adott műholdfelvételhez milyen tanítóadatok tartoznak, illetve azt sem, hogy az adott tanítóadat hol volt mintavételezve. Az alkalmazás eredetileg egy CSV fájlban [39] tárolta el az összes pixel spektrális értékeit és indexeit, és ezt lehetett használni tanításra. Ennek az volt a hátránya, hogy nehéz volt áttekinteni illetve kiegészíteni az adatokat. Ezért az asztali alkalmazást kiegészítettem ezzel a funkcionálitással, a tanítóadatok előállítása elmentésekor az alkalmazás létrehoz egy külön raszteres réteget is külön minden műholdfelvételhez, melyen látható, hogy mely területek voltak hozzáadva a tanítóadatok közé, így tetszőleges módon előállítható/ellenőrizhető a tanítóhalmaz.

#### **5.1.2. A szerveralkalmazás bővítése**

A szerveralkalmazás és webalkalmazás is bővítésre került: a szerveralkalmazás mostmár több modellt is le tud futtatni a letöltött műholdfelvételeken és ezeket külön tárolja. A webalkalmazás mostmár képes letölteni külön ezeket az eredményeket és több hulladékdetektáló módszer eredményét is meg tudja jeleníteni, ennek köszönhetően ezeket egymással össze lehet könnyen hasonlítani valós tesztadatokon [40]. Az 5.1 ábrán látható, ahogy lehet váltakozni a hulladékdetektálási modellek között, és az 5.1 táblázat szerint le-

het beazonosítani a dolgozatban említett modellek<sup>1</sup>. Az old\_model-en kívül az összes model a 3 fejezetben részletezett súlyozással és méretcsökkentési módszerekkel tanítottam be.



5.1. ábra. A webalkalmazás képes több modellnek az eredményét is összehasonlítóképpen megmutatni.

Modell azonosító	Modell leírás
<i>old_model</i>	A Régi modell, amit Magyar és tsai. tanítottak be.
<i>allweather_model</i>	Egy olyan modell, ami kizárolag a romániai hulladéklerakókon volt betanítva.
<i>summer_model</i>	A nyári intervallumban található adatokon tanított modell.
<i>winter_model</i>	A téli intervallumban található adatokon tanított modell.
<i>kiskore_model</i>	A romániai hulladéklerakókon és a kiskörei víztorlaszon tanított modell.
<i>with_extra_unknown_model</i>	Amikor a dolgozatomban az "új modell"-re hivatkozok, erre gondolok. hasonlóan volt betanítva, mint a kiskore_model, annyi különbséggel, hogy hozzávettem több "Ismeretlen" címkehez tartozó adatot, azért, hogy csökkentsem az utak miatt okozott false-positive-okat.

5.1. táblázat. A külön nyári és téli időszakra tanított modellek átlagai.

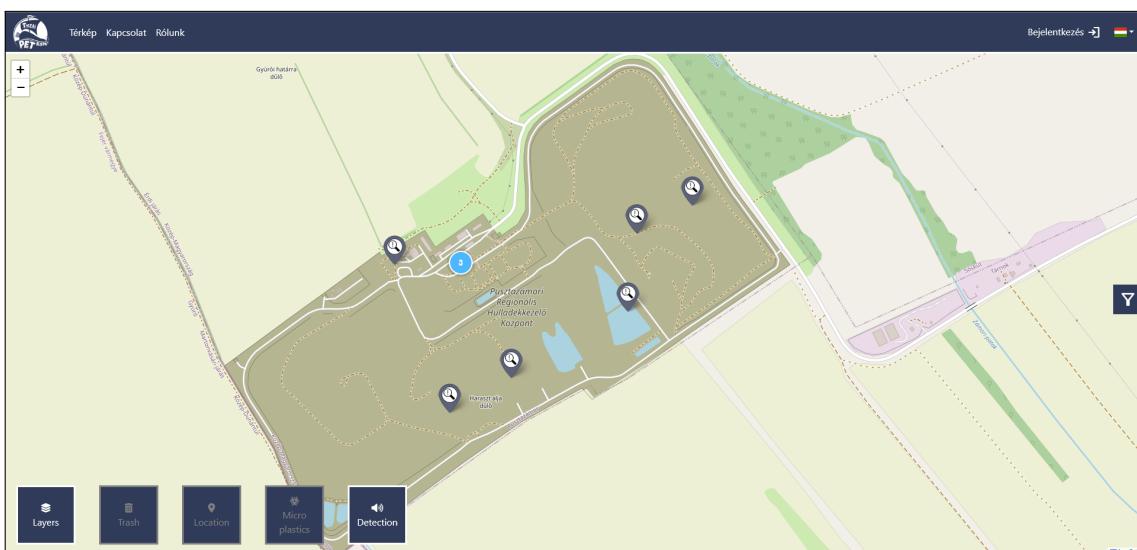
## 5.2. A Tisza-Tisza alkalmazás

A Tisza-Tisza webalkalmazás a PET Kupa egyik webalkalmazása, melynek az a célja, hogy egy olyan felületet biztosítson, ahol meg lehet tekinteni a jelenleg ismert folyómen-

<sup>1</sup>Kivétel erre a főkomponens analízzssel betanított modell, illetve a normalizált felvételeken betanított modell. Ezek várhatóan a jövőben kerülnek fel a weboldalra.

tén található hulladéklerakókat [6]. A PET Kupával együtt dolgozunk azon, hogy ezt az alkalmazást továbbfejlesszük, és a feladataim közé tartozott az is, hogy a Random Forest modell eredményeit integráljam ebbe az alkalmazásba, *backend* oldalon.

Tekintve arra, hogy a Tisza-Tisza térképén pontok vannak megjelenítve, a modell által detektált területeket is pontokkal jelöljük. Ehhez egy nagyobb terület közepére helyezünk el egy pontot. Előfordulhat olyan is, hogy a modell olyan képeket klasszifikál, melyek el vannak torzítva (például magas párata tartalom miatt). Ilyenkor a false-positive-ok aránya lényegesen megnő. Ennek korrigálására a Tisza-Tisza alkalmazásban a legutolsó három detektálást (legfeljebb 1 hónap különbséggel) veszem figyelembe és két kép közös metszetével döntöm el, hogy milyen területek kerülnek fel a térképre. Az 5.2 ábrán láthatóak a hulladékdetectálás során előállított pontok a térképen, melyeket egy nagyítóval vannak jelölve, melyben van egy felkiáltójel. A Hulladékdetectálás konverzióját a térképen látható pontokká az 5.3 fejezetben részletezem.



5.2. ábra. A hulladékdetectálás által megjelenített pontok a térképen.

### 5.2.1. A képfeldolgozás gyorsítása

A meglevő képfeldolgozó algoritmuson gyorsítani kellett, hogy elfogadható időn belül tudja feldolgozni a tanítóadatokat, illetve a naponta letöltött műholdfelvételeket a szerveren. Ezt a kutatólabor korábbi cikkében párhuzamosítással javasolták, de egy egyszálú megoldással lényegesen tudtam gyorsítani a képfeldolgozáson: Ehhez egy nagyon hatékony Python programcsomagot, a Numpy-t [41] használtam fel, mellyel lényegesen megnövelte a feldolgozás sebességét: A tanítóadatok feldolgozásakor a régi módszer (5.1

forráskód) 16 felvételt tudott feldolgozni 4 nap és 10 óra alatt, míg az átírt módszer (5.2 forráskód) feldolgozott 85 felvételt 20 perc alatt.

```
1 def _calculate_index(numerator: np.ndarray, denominator: np.ndarray
2     ) -> np.ndarray:
3     """
4         Calculating an index based on given numerator and denominator.
5
6         :param numerator: numerator matrix
7         :param denominator: denominator matrix
8         :return: result matrix, containing the calculated values
9         """
10
11    # variables
12    rows = numerator.shape[0]
13    cols = numerator.shape[1]
14    index = np.ndarray(
15        shape=numerator.shape,
16        dtype="float32",
17    )
18
19    # calculate index
20    for i in range(rows):
21        for j in range(cols):
22            if np.isnan(numerator[i, j]) or np.isnan(denominator[i, j]):
23                index[i, j] = float("NaN")
24            elif denominator[i, j] != 0:
25                index[i, j] = numerator[i, j] / denominator[i, j]
26            else:
27                if numerator[i, j] < 0:
28                    index[i, j] = np.nanmin(numerator)
29                elif numerator[i, j] > 0:
30                    index[i, j] = np.nanmax(numerator)
31                else:
32                    index[i, j] = float("NaN")
33
34    # return index values
35    return index
```

---

5.1. forráskód. Az index-számolás régi módszere.

```
1 def calculate_index(numerator: np.ndarray, denominator: np.ndarray)
2     -> np.ndarray:
3         """
4             Calculating an index based on given numerator and denominator.
5
6             :param numerator: numerator matrix
7             :param denominator: denominator matrix
8             :return: result matrix, containing the calculated values
9             """
10
11     # variables
12     index = np.ndarray(
13         shape=numerator.shape,
14         dtype="float32",
15     )
16
17     numerator_nan_min = np.nanmin(numerator)
18     numerator_nan_max = np.nanmax(numerator)
19
20     # calculate index
21     nan_mask = np.isnan(numerator) | np.isnan(denominator)
22     numerator_zero_mask = numerator == 0
23     denominator_zero_mask = denominator == 0
24
25     invalid_mask = nan_mask | (numerator_zero_mask &
26                               denominator_zero_mask)
27     valid_mask = np.logical_not(invalid_mask)
28
29     valid_denominator_non_zero_mask = valid_mask & np.logical_not(
30         denominator_zero_mask)
31     valid_denominator_zero_mask = valid_mask &
32         denominator_zero_mask
33
34     numerator_positive_denominator_zero_mask =
35         valid_denominator_zero_mask & (numerator > 0)
36     numerator_negative_denominator_zero_mask =
37         valid_denominator_zero_mask & (numerator < 0)
38
39     index[invalid_mask] = float("NaN")
40     index[numerator_positive_denominator_zero_mask] =
```

```

    numerator_nan_max
35   index[numerator_negative_denominator_zero_mask] =
        numerator_nan_min
36   index[valid_denominator_non_zero_mask] = (
37       numerator[valid_denominator_non_zero_mask] / denominator[
38           valid_denominator_non_zero_mask]
)
39
40   # return index values
41   return index

```

## 5.2. forráskód. Index-számolás numpy-al.

Magyar és tsai. megmérték a modell feldolgozási sebességét különböző méretű felvételeken [5]. Az első felvételt el tudtam kérni a szerzőtől, de a többi felvételt újból előállítottam, igyekezve arra, hogy hasonló legyen a felvételek mérete. Az 5.2 táblázatban találhatóak a régi feldolgozás sebességei a régi modell mellett. Az 5.3 táblázatban látható a régi és új modell feldolgozási sebessége az új számolási módszerrel. Fontos megjegyezni azt, hogy nem ugyanazon a számítógépen történt az új módszer sebességének a lemerése, de a két számítógép számolóképessége ugyanabban a súlycsoporthoz van. A két táblázatból láthatjuk, hogy az új indexszámolási módszer körülbelül felére csökkenette a feldolgozási és osztályozási időt, mostmár az idő nagy része az osztályozással és az osztályozás feldolgozásával megy el. További kutatás esetén érdemes lehet megvizsgálni az osztályozás feldolgozásával kapcsolatos lassulásokat annak érdekében, hogy ezt a feldolgozást is gyorsítsuk.

Felvétel mérete (pixel)	feldolgozási idő
$164 \times 312 = 51\,168$	2.17 mp
$1194 \times 801 = 956\,394$	39.98 mp
$4597 \times 4153 = 19\,091\,341$	12 perc 42 mp
$6614 \times 5981 = 39\,558\,334$	26 perc 5 mp

5.2. táblázat. A régi modell régi módszerrel való feldolgozási ideje.

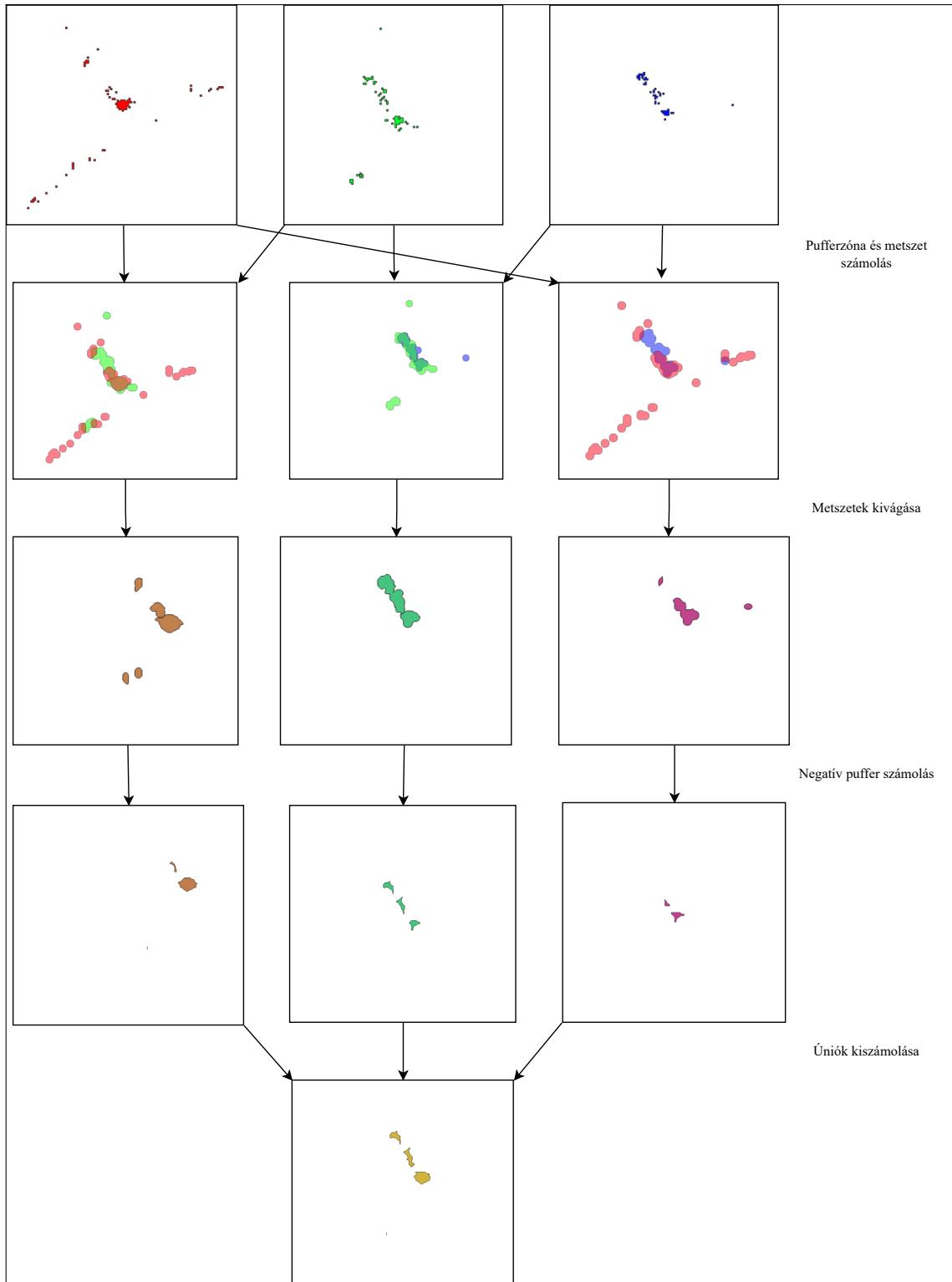
Felvétel mérete (pixel)	Feldolgozási idő (Régi modell)	Feldolgozási idő (Új modell)
$164 \times 312 = 51\ 168$	1.88 mp	2.89 mp
$1262 \times 820 = 1\ 034\ 840$	19.06 mp	13.55 mp
$4951 \times 4002 = 19\ 813\ 902$	6 perc 16 mp	7 perc 41 mp
$10366 \times 3860 = 40\ 012\ 760$	14 perc 24 mp	17 perc 39 mp

5.3. táblázat. A régi és új modell új módszerrel való feldolgozási ideje.

### 5.3. Közös metszet

A már meglevő szervertől poligonok formájában, GeoJSON-ben [42] lehet lekérni az adott napon detektált hulladékos területeket. Így poligonműveletek segítségével lehet ki-számolni az osztályozások közös metszeteit. Az alapötlet az, hogy megfigyeli a modell az adott területet több időpontban, és a több megfigyelésből azokat a területeket jelöljük hulladékosnak, amik a legtöbb felvételen hulladékosoknak voltak megjelölve. Jelöljük  $BUF(P,n)$ -el egy multipolygon pufferét, ahol  $P \in \mathbb{P}$  egy multipolygon, és  $n$  egy egész szám. Ekkor a "többségi szavazást" három időpontban készült felvételre az 5.1 képlet szerint lehet alkalmazni. Ezután a poligonok egy-egy belső pontját megválasztva megtudjuk jelölni a hulladéklerakókat. A megközelítés szavakra bontva a következő: az összes multipolygon párra vesszük a két multipolygon metszetét, (ahol egy  $n$  méretű hibahatár is belefér a metszet számolásba), utána a poligonpárokat rendre összeuniózzuk. Az 5.3 ábra vizualizálja a képlet lépésein. Ezzel a módszerrel ki lehet szűrni azokat a hibákat, melyek rossz minőségű felvételek miatt keletkeznek. Tekintve arra, hogy nem minden Planet felvétel kerül osztályozásra, mivel előfordulnak hiányos vagy felhős felvételek, érdemes megszabni egy időintervallumot, amin belül alkalmazható a többségi szavazás. A dolgozat írásának idejében legfeljebb 1 hónap intervallumon el kell helyezkedjen minden a három felvétel.

$$\text{Eredmény multipolygon} = \bigcup_{P_1 \in \mathbb{P}} \bigcup_{P_2 \in \mathbb{P}} BUF(BUF(P_1, n) \cap BUF(P_2, n), -n) \quad (5.1)$$



5.3. ábra. A 5.1 képlet lépései.

## **6. fejezet**

# **Összefoglalás és eredmények**

### **6.1. A kutatás során elért eredmények**

A dolgozatomban megvizsgáltam több módszert, amivel a korábbi modellben levő ki-hívásokat korrigáltam a multispektrális Planetscope felvételeken. Előállítottam egy tanító adathalmazt, melynek segítségével meg lehet vizsgálni több gépi tanulási módszert, illetve ki lehet próbálni több képfeldolgozási módszert. Megmutattam, hogy egy nagyobb tanító halmaz segítségével alacsonyabb false-positive arányokkal tudja a Random Forest modell detektálni a hulladékkel szennyezett területeket a teszthalmazban. Kiegészítettem az ELTE IK Térinformatikai laborban használt eszközöket arra, hogy hatékonyabban lehessen előkészíteni és megvizsgálni a különböző hulladékdetectálási módszereket. Felgyorsítottam a meglevő index-számolási algoritmust, annak érdekében, hogy hatékonyabban fel lehessen dolgozni a tanítóadatokat. Megvizsgáltam a főkomponens analízis hatását a Random Forestre, és arra a következtetésre jutottam, hogy a modell láthatóan jobban kezelte PCA segítségével a többdimenziós felvételekben levő zajt, mint a PCA nélküli modell. Továbbá megnéztem, hogy miként teljesít a modell képnormalizálás, illetve vízmaszkolás mellett. Az utóbbi két módszert a laborban kutatják és fejlesztik a kollégáim. Bemutattam a módszer használhatóságát azzal, hogy a Tiszta-Tisza webalkalmazásába integráltam a Random Forest modell eredményeit, egyszerű poligonműveletek segítségével.

## 6.2. A kutatás kihívásai

A kutatás talán legnagyobb kihívása a tanító adatok megfelelő előállítása és a modellek validációja. A 2.1 fejezetben bemutatott hulladékdetectálási módszerekben gyakori módszer volt egy magas felbontású műholdfelvétel használata validációra, vagy nagyon magas felbontású drónfelvételek használata a vizsgált területeken. A modell vizsgálata téli felvételeken is egy kihívás tekintve arra, hogy télen ritkábbak a megfelelő időjárás körülmények a nyári időszakokhoz képest.

## 6.3. További lépések

A továbbiakban érdemes megvizsgálni akár személyesen, akár magas felbontású felvételekből a hulladékkal szennyezett területeket pontosabb validáció érdekében. Az adat-normalizálási módszereket is érdemes továbbvizsgálni, tekintve arra, hogy gyakori a zaj a műholdfelvételekben, így ilyen módszerek lényegesen növelhetik a modell megbízhatóságát. További lépésként javasolt akár más gépi tanulási módszereket is kipróbálni, illetve klasszikusabb térinformatikai eszközökkel is megközelíteni a hulladékdetectálás problémáját. Gépi tanulási irány esetén érdemes egy sokkal nagyobb adathalmazt előállítani, pontosabb adatvizsgálat mellett, így a modellek jobban tudják majd általánosítani az adatokat hulladékdetectálás céljából.

# Köszönetnyilvánítás

Köszönetet szeretnék mondani az ELTE Informatikai Karának a TDK ösztöndíj biztosításáért, illetve a PET Kupának a projekt szakmai támogatásáért. Külön köszönet jár a témavezetőmnek, Cserép Máténak, aki szakmailag eligazított a kutatásomban. Köszönetet szeretnék mondani az ELTE IK Térinformatika Laborban levő munkatársaimnak, akikkel együtt dolgoztunk, és akik nélkül nem tartana itt a kutatás. Köszönetet szeretnék mondani a Planet Labs-nak is, akik kutatási célból ingyenesen biztosítottak műholdfelvételeket a labor számára.

# Irodalomjegyzék

- [1] M.G. Kibria, N.I. Masuk és R. et al. Safayet. „Plastic Waste: Challenges and Opportunities to Mitigate Pollution and Effective Management”. *International Journal of Environmental Research* 17.20 (2023. jan.). ISSN: 2008-2034. URL: <https://doi.org/10.1007/s41742-023-00507-z>.
- [2] Björn Nyberg és tsai. „Leaving a plastic legacy: Current and future scenarios for mismanaged plastic waste in rivers”. *Science of The Total Environment* 869 (2023), 161821. old. ISSN: 0048-9697. DOI: <https://doi.org/10.1016/j.scitotenv.2023.161821>. URL: <https://www.sciencedirect.com/science/article/pii/S0048969723004369>.
- [3] Tim H. M. van Emmerik és tsai. „River plastic transport and deposition amplified by extreme flood”. *Nature Water* 1.6 (2023), 514–522. old. ISSN: 2731-6084. DOI: [10.1038/s44221-023-00092-7](https://doi.org/10.1038/s44221-023-00092-7). URL: <https://doi.org/10.1038/s44221-023-00092-7>.
- [4] Leo Breiman. „Random Forests”. *Machine Learning* 45.1 (2001), 5–32. old. ISSN: 1573-0565. DOI: [10.1023/A:1010933404324](https://doi.org/10.1023/A:1010933404324). URL: <https://doi.org/10.1023/A:1010933404324>.
- [5] Dávid Magyar és tsai. „Waste Detection and Change Analysis based on Multispectral Satellite Imagery”. 2023. jan. DOI: [10.48550/arXiv.2303.14521](https://arxiv.org/abs/2303.14521).
- [6] PET Kupa. URL: <https://tisztatiszaterkep.hu/> (elérés dátuma 2024. 05. 11.).
- [7] Anjar Dimara Sakti és tsai. „Identification of illegally dumped plastic waste in a highly polluted river in Indonesia using Sentinel-2 satellite imagery”. *Scientific Reports* 13.1 (2023), 5039. old. ISSN: 2045-2322. DOI: [10.1038/s41598-023-32087-5](https://doi.org/10.1038/s41598-023-32087-5). URL: <https://doi.org/10.1038/s41598-023-32087-5>.

- [8] Gil Gonçalves és Umberto Andriolo. „Operational use of multispectral images for macro-litter mapping and categorization by Unmanned Aerial Vehicle”. *Marine Pollution Bulletin* 176 (2022), 113431. old. ISSN: 0025-326X. DOI: <https://doi.org/10.1016/j.marpolbul.2022.113431>. URL: <https://www.sciencedirect.com/science/article/pii/S0025326X22001138>.
- [9] Antonio Lanorte és tsai. „Agricultural plastic waste spatial estimation by Landsat 8 satellite images”. *Computers and Electronics in Agriculture* 141 (2017), 35–45. old. ISSN: 0168-1699. DOI: <https://doi.org/10.1016/j.compag.2017.07.003>. URL: <https://www.sciencedirect.com/science/article/pii/S0168169917302259>.
- [10] Dan Zeng és tsai. „Multi-Scale CNN Based Garbage Detection of Airborne Hyperspectral Data”. *IEEE Access* 7 (2019), 104514–104527. old. DOI: 10.1109/ACCESS.2019.2932117.
- [11] Xian Sun és tsai. „Revealing influencing factors on global waste distribution via deep-learning based dumpsite detection from satellite imagery”. *Nature Communications* 14.1 (2023), 1444. old. ISSN: 2041-1723. DOI: 10.1038/s41467-023-37136-1. URL: <https://doi.org/10.1038/s41467-023-37136-1>.
- [12] Rocio Nahime Torres és Piero Fraternali. „Learning to Identify Illegal Landfills through Scene Classification in Aerial Images”. *Remote Sensing* 13.22 (2021). ISSN: 2072-4292. DOI: 10.3390/rs13224520. URL: <https://www.mdpi.com/2072-4292/13/22/4520>.
- [13] Kaiming He és tsai. „Deep Residual Learning for Image Recognition”. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.
- [14] Tsung-Yi Lin és tsai. „Feature Pyramid Networks for Object Detection”. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.
- [15] Robert Page és tsai. „Identification of Tyre and Plastic Waste from Combined Copernicus Sentinel-1 and -2 Data”. *Remote Sensing* 12.17 (2020). ISSN: 2072-4292. DOI: 10.3390/rs12172824. URL: <https://www.mdpi.com/2072-4292/12/17/2824>.

- [16] Nicolò Taggio és tsai. „A Combination of Machine Learning Algorithms for Marine Plastic Litter Detection Exploiting Hyperspectral PRISMA Data”. *Remote Sensing* 14.15 (2022). ISSN: 2072-4292. DOI: 10 . 3390 / rs14153606. URL: <https://www.mdpi.com/2072-4292/14/15/3606>.
- [17] Mattis Wolf és tsai. „Machine learning for aquatic plastic litter detection, classification and quantification (APLASTIC-Q)”. *Environmental Research Letters* 15.11 (2020), 114042. old. DOI: 10 . 1088 / 1748 - 9326 / abbd01. URL: <https://doi.org/10.1088/1748-9326/abbd01>.
- [18] Gil Gonçalves és tsai. „Mapping marine litter using UAS on a beach-dune system: a multidisciplinary approach”. *Science of The Total Environment* 706 (2020), 135742. old. ISSN: 0048-9697. DOI: <https://doi.org/10.1016/j.scitotenv.2019.135742>. URL: <https://www.sciencedirect.com/science/article/pii/S0048969719357377>.
- [19] Ousmane Youme és tsai. „Deep Learning and Remote Sensing: Detection of Dumping Waste Using UAV”. *Procedia Computer Science* 185 (2021). Big Data, IoT, and AI for a Smarter Future, 361–369. old. ISSN: 1877-0509. DOI: <https://doi.org/10.1016/j.procs.2021.05.037>. URL: <https://www.sciencedirect.com/science/article/pii/S1877050921011224>.
- [20] Barry de Ville. „Decision trees”. *WIREs Computational Statistics* 5.6 (2013), 448–455. old. DOI: <https://doi.org/10.1002/wics.1278>. eprint: <https://wires.onlinelibrary.wiley.com/doi/pdf/10.1002/wics.1278>. URL: <https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.1278>.
- [21] Jr Harrel. URL: <https://lib.stat.cmu.edu/S/Harrell/data/descriptions/titanic.html> (elérés dátuma 2013. 09. 23.).
- [22] IBM. URL: <https://www.ibm.com/topics/random-forest> (elérés dátuma 2024. 05. 05.).
- [23] Planet Labs. URL: <https://www.planet.com/company/> (elérés dátuma 2024. 05. 06.).
- [24] European Space Agency. URL: <https://earth.esa.int/eogateway/missions/planetscope> (elérés dátuma 2024. 05. 06.).
- [25] Planet Labs. URL: <https://developers.planet.com/docs/apis/data/sensors/> (elérés dátuma 2024. 04. 03.).

- [26] InfoCons.ro. URL: <https://fiiunexemplu.ro/in-romania-exista-46-depozite-de-deseuri-gropi-de-gunoii> (elérés dátuma 2024. 04. 02.).
- [27] Google. URL: <https://www.google.com/maps/> (elérés dátuma 2024. 05. 01.).
- [28] Pet Kupa. URL: <https://www.petkupa.hu/> (elérés dátuma 2024. 05. 08.).
- [29] Kyriacos Themistocleous és tsai. „Investigating Detection of Floating Plastic Litter from Space Using Sentinel-2 Imagery”. *Remote Sensing* 12.16 (2020). ISSN: 2072-4292. DOI: 10.3390/rs12162648. URL: <https://www.mdpi.com/2072-4292/12/16/2648>.
- [30] S. K. McFEETERS. „The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features”. *International Journal of Remote Sensing* 17.7 (1996), 1425–1432. old. DOI: 10.1080/01431169608948714.
- [31] F. Pedregosa és tsai. „Scikit-learn: Machine Learning in Python”. *Journal of Machine Learning Research* 12 (2011), 2825–2830. old.
- [32] Hervé Abdi és Lynne J. Williams. „Principal component analysis”. *WIREs Computational Statistics* 2.4 (2010), 433–459. old. DOI: <https://doi.org/10.1002/wics.101>. eprint: <https://wires.onlinelibrary.wiley.com/doi/pdf/10.1002/wics.101>. URL: <https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.101>.
- [33] István Elek. *Adatbázisok, térképek, információs rendszerek*. ELTE Eötvös kiadó, 2011. ISBN: 978-963-312-039-2.
- [34] Mangale Sanchita. URL: <https://sanchitamangale12.medium.com/scree-plot-733ed72c8608> (elérés dátuma 2024. 05. 11.).
- [35] Tom Howley és tsai. „The Effect of Principal Component Analysis on Machine Learning Accuracy with High Dimensional Spectral Data”. 2005. jan., 209–222. old. ISBN: 978-1-84628-223-2. DOI: 10.1007/1-84628-224-1\_16.
- [36] EuroNews. URL: <https://www.euronews.com/green/2023/06/04/we-will-die-before-the-drina-is-clean-bosnia-villagers-hope-for-a-solution-to-polluted-riv> (elérés dátuma 2024. 05. 05.).

- [37] Russell G. Congalton. „A review of assessing the accuracy of classifications of remotely sensed data”. *Remote Sensing of Environment* 37.1 (1991), 35–46. old. ISSN: 0034-4257. DOI: [https://doi.org/10.1016/0034-4257\(91\)90048-B](https://doi.org/10.1016/0034-4257(91)90048-B). URL: <https://www.sciencedirect.com/science/article/pii/003442579190048B>.
- [38] Anett Fekete és Mate Cserep. „Tree segmentation and change detection of large urban areas based on airborne LiDAR”. *Computers & Geosciences* 156 (2021), 104900. old. ISSN: 0098-3004. DOI: <https://doi.org/10.1016/j.cageo.2021.104900>. URL: <https://www.sciencedirect.com/science/article/pii/S0098300421001928>.
- [39] Yakov Shafranovich. *Common Format and MIME Type for Comma-Separated Values (CSV) Files*. RFC 4180. 2005. okt. DOI: 10.17487/RFC4180. URL: <https://www.rfc-editor.org/info/rfc4180>.
- [40] ELTE IK Térinformatikai Labor. URL: <https://gis.inf.elte.hu/waste-detection/> (elérés dátuma 2024. 05. 11.).
- [41] Charles R. Harris és tsai. „Array programming with NumPy”. *Nature* 585.7825 (2020. szept.), 357–362. old. DOI: 10.1038/s41586-020-2649-2. URL: <https://doi.org/10.1038/s41586-020-2649-2>.
- [42] H. Butler és tsai. *The GeoJSON Format*. RFC 7946. 2016. aug. DOI: 10.17487/RFC7946. URL: <https://www.rfc-editor.org/info/rfc7946>.

# Ábrák jegyzéke

2.1. Hulladékdetektálás Adjusted Plastic Index, Random Forest és Mahalanobis távolság segítségével [7]. . . . .	7
2.2. Spectral angle mapping referencia adatainak előállítása [8]. . . . .	8
2.3. Az MSCNN és NMS működési elve [10]. . . . .	9
2.4. A BCA-Net működési elve [11]. . . . .	10
2.5. A tengeri hulladékdetektáláshoz való tanító adatok előállítása [16]. . . . .	11
2.6. A gépi tanulás, geomorfológia, fotogrammetria és hidrodinamikai modellezés használása hulladékdetektálásra [18]. . . . .	12
2.7. A döntési fa működési elve [20]. . . . .	13
2.8. A Random Forest működési elve [22]. . . . .	14
3.1. A kiskörei víztároló hulladékторlasza [28]. . . . .	17
3.2. A műanyag alapú hulladék, törmelék mellé helyezve. Forrás: Google Maps.	18
3.3. Különböző indexek összehasonlítása egy mesterségesen előállított úszó műanyag-palack szigeten [29]. . . . .	20
3.4. A csökkentett modell keveseb <i>true positive</i> értéket termel a teljes méretű modellhez képest a pusztazámi hulladéklerakó egyik felvételén. Felvétel dátuma: 2021.12.08. . . . .	21
3.5. Az adatok közötti aránytalanság, logaritmikus skálázással. . . . .	22
3.6. A főkomponensek varianciája a tanító halmazon. 90% variancia megtartásának érdekében elég az első három főkomponenst kiválasztani. . . . .	23
3.7. A főkomponens analízis geometriai jelentése: a standardizálás 0 várható értékűvé és 1 empirikus szórásúvá teszi a változókat, vagyis a pontfelhőt betolja az origóba, majd elforgatja a legnagyobb variancia irányába, ami az első főkomponens [33]. . . . .	24
3.8. Nyári és téli adatok összehasonlítása. . . . .	25

4.1.	A drinai hulladéksziget. Egy lebegő zsinór fogja meg a műanyagpalackokat [36]. . . . .	27
4.2.	A Drina melletti féllegális szemétlerakó [28]. . . . .	27
4.3.	Az új modell összehasonlítása a régi modellel az egyik Drinai teszt felvételen. Felvétel dátuma: 2023.05.07. . . . .	28
4.4.	Az új modell összehasonlítása a PCA-val tanított modellel az egyik Drinai teszt felvételen. Felvétel dátuma: 2023.05.07. . . . .	30
4.5.	A főkomponens analízissel betanított modell osztályozásának összehasonlítása a PCA nélkül betanított modellel. Felvétel dátuma: 2023.05.21. .	31
4.6.	A vízmaszkolt osztályozás összehasonlítása a vízmaszk nélküli osztályozással. A Drina melletti hulladéklerakó kizáráusra kerül. Felvétel dátuma: 2023.05.07. . . . .	32
4.7.	Az új modell összehasonlítása a nyári modellel az egyik Drinai teszt felvételen. Felvétel dátuma: 2023.05.07. . . . .	34
4.8.	A régi, a normalizáció nélkül tanított és a normalizációval tanított modell összehasonlítása a Drinán. Felvétel dátuma: 2023.05.24. . . . .	35
4.9.	Az összes modell összehasonlítása az egyik téli drinai teszt felvételen. Felvétel dátuma: 2023.12.17. . . . .	37
4.10.	Az összes modell összehasonlítása az egyik kiskörei teszt felvételen. Felvétel dátuma: 2024.04.12. . . . .	38
5.1.	A webalkalmazás képes több modellnek az eredményét is összehasonlítóképpen megmutatni. . . . .	40
5.2.	A hulladékdetectálás által megjelenített pontok a térképen. . . . .	41
5.3.	A 5.1 képlet lépései. . . . .	46

# Táblázatok jegyzéke

2.1.	A PSB.SD műszer hullámhosszai [25]. . . . .	15
3.1.	A tanító adatok címkéi. . . . .	18
3.2.	A döntési fák méretének a mediánja nem csökkent, amint a főkomponensek száma csökkent, cserében 3 főkomponensnél már nőtt. . . . .	24
4.1.	A régi modell és az új modell teszteredményei átlagolva. . . . .	29
4.2.	A főkomponens analízissel betanított modell teljesítményének az átlagai. .	31
4.3.	A nyári időszakra tanított modell átlagai. . . . .	33
5.1.	A külön nyári és téli időszakra tanított modellek átlagai. . . . .	40
5.2.	A régi modell régi módszerrel való feldolgozási ideje. . . . .	44
5.3.	A régi és új modell új módszerrel való feldolgozási ideje. . . . .	45